"THE MATERIAL HEREIN IS FOR INFORMATION PURPOSES ONLY AND IS SUBJECT TO CHANGE WITHOUT NOTICE. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS WHICH MAY APPEAR HEREIN."

FIELD MAINTENANCE PRINT SET

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION. COPYRIGHT © 1978 DIGITAL EQUIPMENT CORPORATION."

TABLE OF CONTENTS

B-TC-11/Ø3-N-1	11/Ø3-N FIELD MAINTENANCE PRINT SET
E-UA-11/Ø3-N-Ø	11/Ø3-N UNIT ASSY
A-PL-11/Ø3-N-Ø	11/Ø3-N UNIT ASSY (PL)
E-UA-BA11-R-Ø	BA11-R UNIT ASSY
A-PL-BA11-R-Ø	BA11-R UNIT ASSY (PL)
D-CS-M7264-Ø-1	LSI-11 CPU MODULE
D-CS-M7955-Ø-1	MSV11-C LSI MOS MEMORY
D-CS-M8Ø44-Ø-1	32K 16 BIT MOS MEMORY
D-CS-M8Ø13-Ø-1	RLV11 DISK CONTROL
D-CS-M8Ø14-Ø-1	RLV11 BUS CONTROL
D-CS-M8Ø29-Ø-1	RXV21 LSI11 RXØ CONTROLLER
D-CS-M794Ø-Ø-1	SERIAL LINE UNIT
D-CS-M8Ø27-Ø-1	LPV11 PRINTER INTERFACE
K-CS-BDV11-A-2	DIAGNOSTIC/BOOT ROM LISTING

UNIT VARIATIONS COVERED BY THIS PRINT SET
11/Ø3-NA
11/Ø3-NB
11/Ø3-NC
11/Ø3-ND

11/Ø3-N

Field Maintenance Print Set

Digital Equipment Corporation

PRINT SET ORDER NO.
MPØØ638

REV.

SIZE B | CODE TC | NUMBER 11/Ø3-N-1

REVISIONS	REV		
	CHG. NO.		
	DATE		

USED ON OPTION/MODEL	
11/Ø3-N	

SHEET 1 OF 1

DRN. R. ROBICHAUD	DATE 21APR78	digital
CHK'D J. FALKOWSKI	DATE 6-JUNE 78	TITLE: 11/Ø3-N TABLE OF CONTENT
PROJ. ENG.	DATE	
FIELD SERV.	DATE	SIZE B / CODE TC / NUMBER 11/Ø3-N-1 / REV. / DIST.

DRB 124

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 19__ DIGITAL EQUIPMENT CORPORATION"

WIRE TABLE

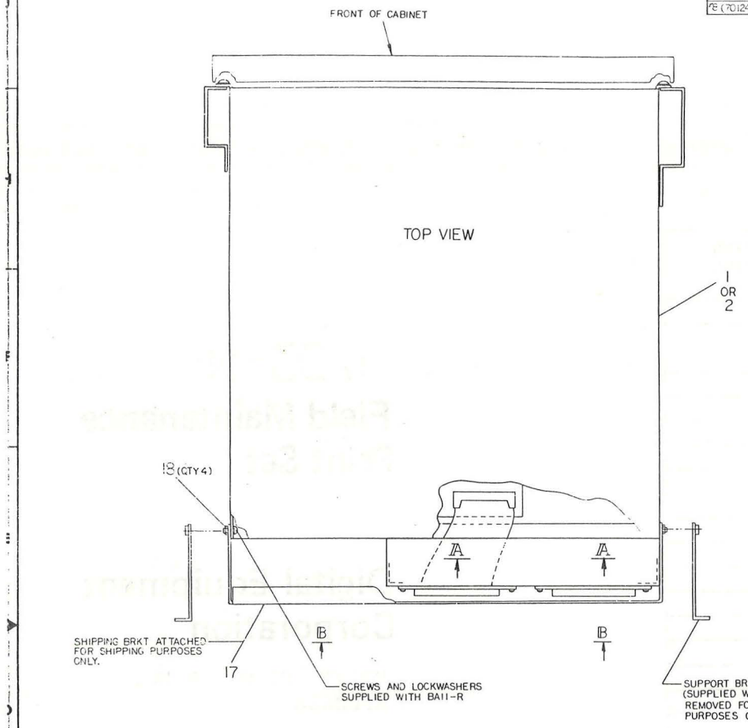
ITEM NO.	FROM	TO	USED ON THIS TYPE OF SYSTEM
#3 (BC03L-1K)	SLOT J5	J1 OF M7940 (ITEM #12)	RLØ1 & RXØ2
#4 (BC26K-1K)	SLOT J3	J1 OF M8027 (ITEM #11)	RL21 & RXØ2
#5 (7012405-00)	SLOT J8	J1 OF M8013 (PART OF ITEM #6)	RL21 ONLY
#6 (7012405-00)	SLOT J7	J1 OF M8029 (PART OF ITEM #7)	RLØ1 ONLY

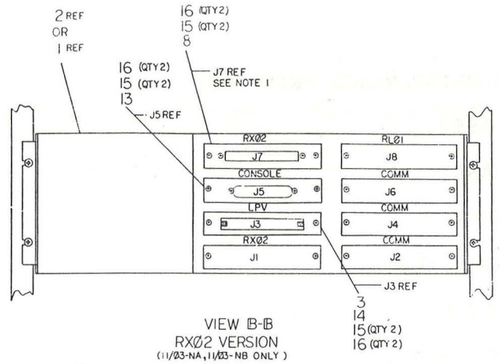
LEGEND

NUMBER	VARIATION	USED WITH
11/03-NA	115V, 60HZ, 12 AMP	RXØ2
11/03-NB	230V, 50HZ, 6 AMP	RXØ2
11/03-NC	115V, 50HZ, 12 AMP	RLØ1
11/03-ND	230V, 50HZ, 6 AMP	RLØ1

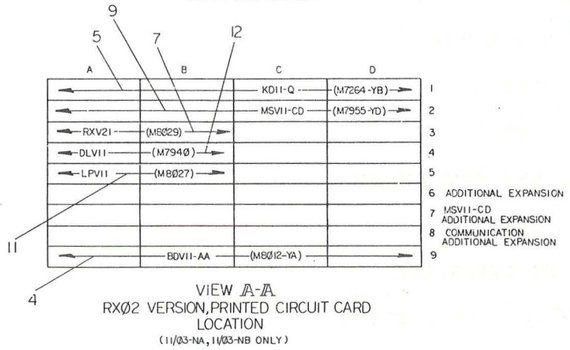
NOTES:

1. ATTACH ONE END OF CABLE (ITEM #3) TO CONNECTOR PANEL AT REAR OF ITEM #1 OR #2 AS SHOWN AND THE OTHER END TO J1 OF PRINTED CIRCUIT CARD AS INDICATED IN WIRE TABLE.
2. ATTACH ONE END OF CABLE (ITEM #4) TO M7224-B FILTER CONNECTOR (ITEM #3) AND THEN TO BE CONNECTED TO CONNECTOR PANEL AT REAR OF ITEM #1 OR #2 AS SHOWN. OTHER END OF CABLE (ITEM #4) TO J1 OF PRINTED CIRCUIT CARD AS INDICATED IN WIRE TABLE.
3. FOR CABINET MOUNTING INSTRUCTIONS FOR 11/03-N, SEE DRAWING: E-UA-BA11-B-Z (SHEET 2).

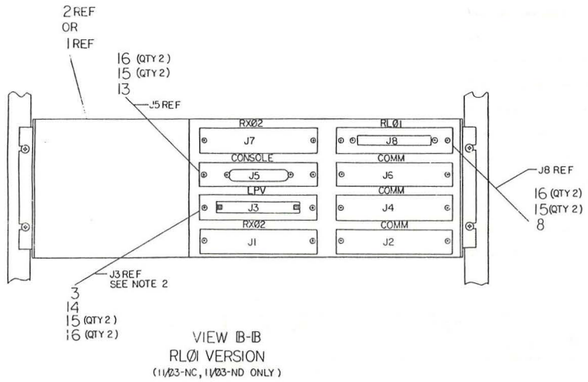




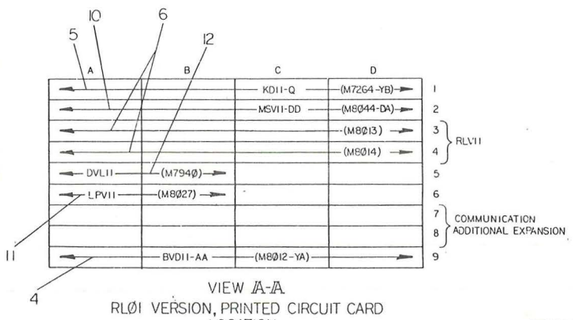
VIEW B-B
RXØ2 VERSION
(11/03-NA, 11/03-NB ONLY)



VIEW A-A
RXØ2 VERSION, PRINTED CIRCUIT CARD LOCATION
(11/03-NA, 11/03-NB ONLY)



VIEW B-B
RLØ1 VERSION
(11/03-NC, 11/03-ND ONLY)



VIEW A-A
RLØ1 VERSION, PRINTED CIRCUIT CARD LOCATION
(11/03-NC, 11/03-ND ONLY)

CAUTION:
CFF SHEET PARTS LIST REFER TO A-PL-11-03-N-0

QUANTITY & VARIATION	DESCRIPTION	DWG./PART NO.	ITEM NO.
	UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		
THIRD ANGLE PROJECTION	DRWN		FIRST USED ON 103
REMOVE BURRS AND BREAK SHARP CORNERS	CHK'D		TITLE: 11/03-N UNIT ASSY
DO NOT SCALE DWG	ENG.		
MATERIAL: SEE PARTS LIST	PROJ. ENG.		
FINISH	PROD.		
	NEXT HIGHER ASSY.	3-2511-03-N	SIZE E, CODE UA, NUMBER 11/03-N-0, REV.
	SCALE NONE		
	SHEET OF 1		DIST.

REVISIONS: REV. | CHANGE NO. | CHK

MK

2

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS
PARTS LIST

MADE BY R. ROBICHAUD	CHECKED J. FALKOWSKI	SECTION
DATE 28-APR-78	DATE 4-MAY-78	1
ENG	PROD	ISSUED SECT.
DATE	DATE	1

ITEM NO.	DWG NO./PART NO.	DESCRIPTION	QUANTITY VARIATION 11/Ø3-NA	11/Ø3-NB	11/Ø3-NC	11/Ø3-ND
1	E-UA-BA11-RA-Ø	BA11-R UNIT ASSY 115V,5ØHZ/6ØHZ 12 AMP	1	-	1	-
2	E-UA-BA11-RB-Ø	BA11-R UNIT ASSY 23ØV,5ØHZ/6ØHZ 6 AMP	-	1	-	1
3	A-PL-H7ØØ4-B-Ø	H7ØØ4 FILTER CONNECTOR	1	1	1	1
4	D-UA-BDV11-AA-Ø	UNIT ASSY (BDV11)	1	1	1	1
5	C-UA-KD11-Q-Ø	LSI-11 PROCESSOR & KEV11-CA	1	1	1	1
6	A-PL-RLV11-Ø-Ø	RLV11 DISK/BUS CONTROL	-	-	1	1
7	A-PL-RXV21-Ø-Ø	RXØ2 DISK INTERFACE (M8029)	1	1	-	-
8	C-AD-7012405-0-0	I/O CABLE ASSY W/MTG BRKT	1	1	1	1
9	A-PL-MSV11-C-Ø	16K MOS RAM	1	1	-	-
10	A-PL-MSV11-D-Ø	MSV11-D (32K-16) MOS MEMORY SYSTEM	-	-	1	1
11	E-UA-LPV11-Ø-Ø	PRINTER INTERFACE	1	1	1	1
12	A-PL-DLV11-Ø-Ø	SERIAL LINE UNIT	1	1	1	1
13	D-UA-BCØ3L-1K-Ø	FILTERED CABLE ASSY (BCØ3L)	1	1	1	1
14	D-UA-BCØ6K-1K-Ø	MIRROR IMAGE CABLE	1	1	1	1
15	9006021-01	SCR PHL,PAN HD 6-32x.31 LG	6	6	6	6
16	9006633-00	WASHER INT. TOOTH LOCK #6	6	6	6	6
17	D-MD-7417680-0-0	BRACKET,SHIPPING, 11/Ø3-N	1	1	1	1
18	9006563-00	NUT,KEPS #8	4	4	4	4

TITLE	ASSY NO.	SIZE	CODE	NUMBER	REV	ECO NO.
UNIT ASSY,11/Ø3-N	D-LD-11/Ø3-Ø	A	PL	11/Ø3-N-Ø		
	SHEET 1 OF 1					

DEC FORM
DRA 110

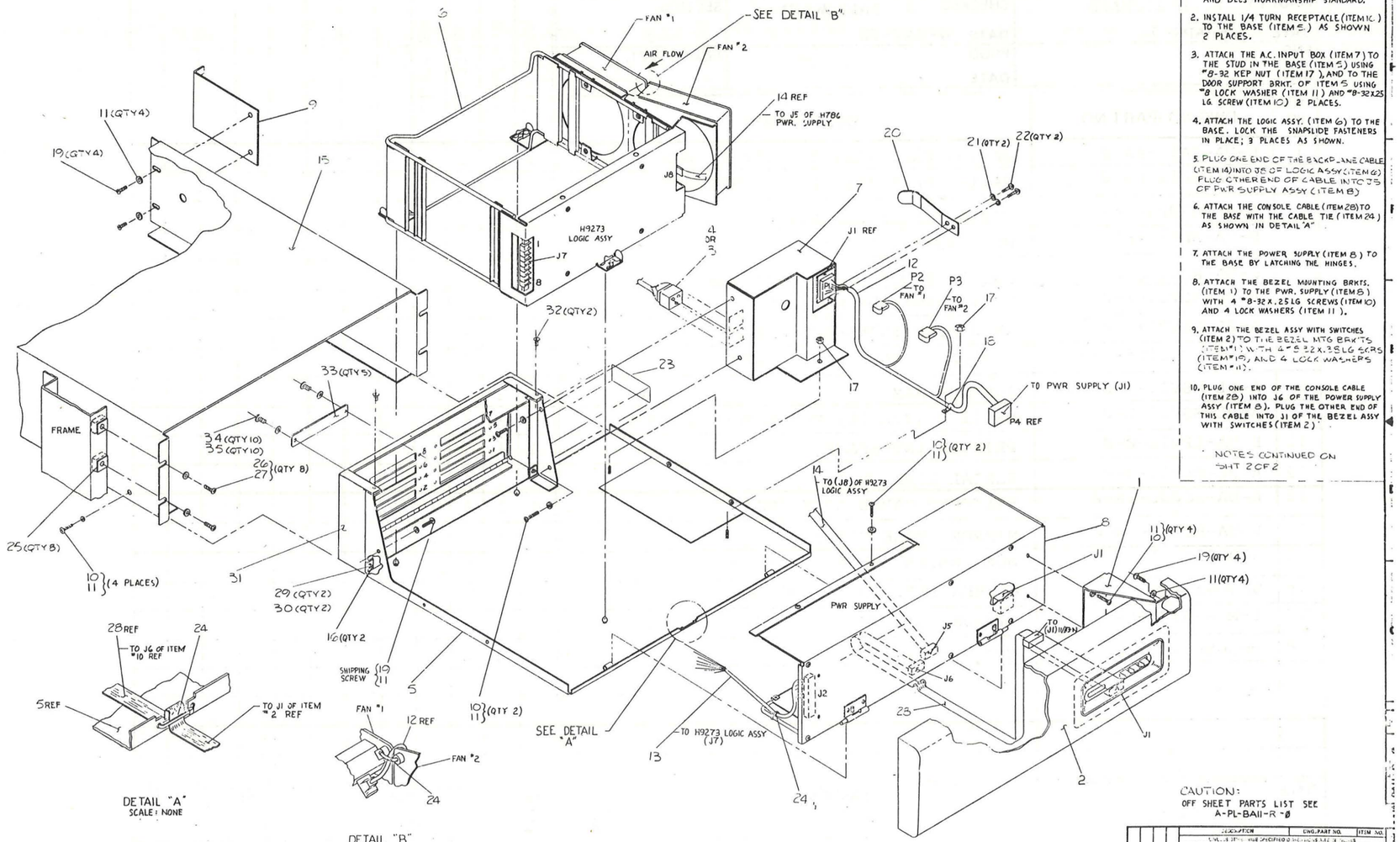
LEGEND					
NUMBER	VARIATION				
	AMPS	PHASE/WIRES	VOLTS	HZ	
BA11-RA	12A	SINGLE/3	115V	50/60HZ	BEZEL ASSY/ SWITCHES
BA11-RB	6A	SINGLE/3	230V	50/60 HZ	BEZEL ASSY/ SWITCHES

NOTES:

ASSEMBLY INSTRUCTIONS

1. OPERATIONS TO BE PERFORMED PER HARDWARE STANDARDS NO. SP-7665099-0 AND DECS WORKMANSHIP STANDARD.
2. INSTALL 1/4 TURN RECEPTACLE (ITEM 6.) TO THE BASE (ITEM 5.) AS SHOWN 2 PLACES.
3. ATTACH THE A.C. INPUT BOX (ITEM 7) TO THE STUD IN THE BASE (ITEM 5) USING #8-32 KEP NUT (ITEM 17.), AND TO THE DOOR SUPPORT BRKT. OF ITEM 5 USING #8 LOCK WASHER (ITEM 11) AND #8-32X.25 LG SCREW (ITEM 10) 2 PLACES.
4. ATTACH THE LOGIC ASSY. (ITEM 6) TO THE BASE. LOCK THE SNAPSLIDE FASTENERS IN PLACE; 3 PLACES AS SHOWN.
5. PLUG ONE END OF THE BACKPLANE CABLE (ITEM 14) INTO J8 OF LOGIC ASSY (ITEM 6) PLUG OTHER END OF CABLE INTO J5 OF PWR SUPPLY ASSY (ITEM 8)
6. ATTACH THE CONSOLE CABLE (ITEM 28) TO THE BASE WITH THE CABLE TIE (ITEM 24.) AS SHOWN IN DETAIL "A"
7. ATTACH THE POWER SUPPLY (ITEM 8) TO THE BASE BY LATCHING THE HINGES.
8. ATTACH THE BEZEL MOUNTING BRKTS. (ITEM 1) TO THE PWR. SUPPLY (ITEM 8) WITH 4 #8-32 X .25 LG SCREWS (ITEM 10) AND 4 LOCK WASHERS (ITEM 11).
9. ATTACH THE BEZEL ASSY WITH SWITCHES (ITEM 2) TO THE BEZEL MTG BRKTS (ITEM 1) WITH 4 #5-32 X .375 LG SCRS (ITEM 10) AND 4 LOCK WASHERS (ITEM 11).
10. PLUG ONE END OF THE CONSOLE CABLE (ITEM 28) INTO J6 OF THE POWER SUPPLY ASSY (ITEM 8). PLUG THE OTHER END OF THIS CABLE INTO J1 OF THE BEZEL ASSY WITH SWITCHES (ITEM 2)

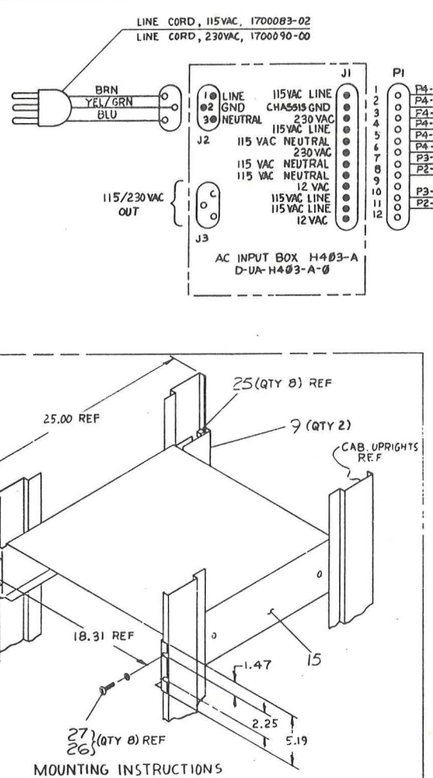
NOTES CONTINUED ON SHT 2 OF 2



CAUTION:
OFF SHEET PARTS LIST SEE A-PL-BA11-R-0

QUANTITY & VARIATION	DESCRIPTION	DWG. PART NO.	ITEM NO.
	UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		
	TOLERANCES: FRACTIONS ±1/32, ANGLES ±1°, DECIMALS: .XX ±.01, .XXX ±.005		
	MATERIAL		
	FINISH		
	DO NOT SCALE DRAWING		
	NEXT ASSY		
	USED ON		
	APPLICATION	DO NOT USE	
	QUALITY		
	DIMENSIONS	ARE IN INCHES	

"THIS DRAWING AND SPECIFICATIONS, WHETHER THE PROPERTY OF [illegible] SHALL NOT BE REPRODUCED IN WHOLE OR IN PART WITHOUT THE WRITTEN PERMISSION OF [illegible] COPYRIGHT © 1978 DIGITAL EQUIPMENT CORPORATION"



MOUNTING INSTRUCTIONS

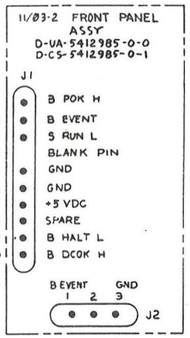
1. REMOVE THE 4 SCREWS (ITEM #10 THAT ATTACH THE COVER TO THE BASE. THEY ARE FOR NON RACK MOUNTED APPLICATIONS.
2. OPEN THE FILTER HSG (ITEM #31) AND REMOVE THE SHIPPING SCREW (ITEM #19).
3. SLIDE THE COVER (ITEM #15) OFF THE UNIT.
4. ATTACH THE TINNERMAN NUTS TO THE CABINET UPRIGHTS 8 PLACES AS INDICATED ABOVE.
5. MOUNT THE COVER (ITEM #15) TO THE FRONT CABINET UPRIGHTS USING 4 PHL. TRUSS HD SCREWS #10-32 X .50 LG. (ITEM #27) AND 4 #10 FLAT WASHERS (ITEM #26)
6. ATTACH THE 2 SUPPORT BRACKETS (ITEM #9) TO THE COVER USING 4 PHL. PAN HD. SCREWS #8-32 X .38 LG. (ITEM #19) AND 4 #8 LOCK WASHERS (ITEM #11).
7. ATTACH THE SUPPORT BRACKETS (ITEM #9) TO THE REAR CABINET UPRIGHTS USING 4 PHL. TRUSS HD SCREWS #10-32 X .50 LG. (ITEM #27) AND 4, #10 FLAT WASHERS (ITEM #26).
8. SLIDE THE UNIT INTO THE COVER. IT WILL BE HELD IN PLACE BY THE SPRING BUTTON (ITEM #20). TO SLIDE THE UNIT FORWARD AGAIN IT WILL BE NECESSARY TO RELEASE THIS SPRING BUTTON.
9. IF THE SYSTEM IS TO BE MOVED OR SHIPPED THE SHIPPING SCREW (ITEM #19) MUST BE REPLACED.

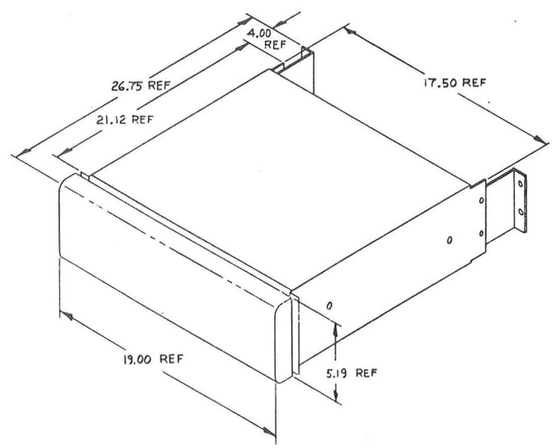
ASSEMBLY INSTRUCTIONS (CONT)

11. LOOSEN THE SCREWS ON CONNECTOR J2 OF THE PWR. SUPPLY ASSY (ITEM #8) AND ATTACH THE FANING STRIP END OF THE D.C. HARNESS (ITEM #13). TORQUE SCREWS TO 10 LBS. IN. ATTACH THE OTHER END OF THIS CABLE BY LOOSENING THE SCREWS ON CONN. J7 OF THE LOGIC ASSY (ITEM #6) AND INSERTING THE END OF EACH WIRE PER THE WIRING DIAGRAM ON SHT #2 OF THIS PRINT. TORQUE THE SCREWS OF CONN J7 TO 10 LBS. IN.
12. SECURE THE D.C. HARNESS (ITEM #13) TO THE POWER SUPPLY WIRE FORM USING ONE TIE WRAP (ITEM #24).
13. ATTACH THE AC. HARNESS (ITEM #12) BY PLUGGING: P1 OF THE HARNESS INTO J1 OF THE AC INPUT BOX (ITEM #7), P4 OF THE HARNESS INTO J1 OF THE PWR. SUPPLY (ITEM #8), P2 OF THE HARNESS INTO FAN NO. 1 OF THE LOGIC ASSY (ITEM #6) AND P3 OF THE HARNESS INTO FAN NO. 2 OF THE LOGIC ASSY (ITEM #6).
14. SECURE THE AC. HARNESS (ITEM #12) TO THE BASE WITH THE CABLE CLAMP (ITEM #18) AND A #8-32 KEPS NUT (ITEM #17), ALSO SECURE THE FAN CORD ATTACHED TO FAN NO.1 WITH A CABLE TIE (ITEM #24). SEE DETAIL "B" ON SHEET #1.
15. CLOSE THE PWR SUPPLY (ITEM #8) AND SECURE IT TO THE LOGIC ASSY (ITEM #6) WITH 2 #8-32 X .25 LG. SCREWS (ITEM #10) AND 2 LOCK WASHERS (ITEM #11).
16. ATTACH THE SPRING BUTTON ASSY (ITEM #20) TO THE AC INPUT (ITEM #7) WITH 2 #4-40 X .19 LG SCWS (ITEM #22) AND 2 LOCK WASHERS (ITEM #21).

ASSY. INSTRUCTIONS (CONT)

17. SLIDE THE COVER (ITEM #15) OVER THE ASSY AND SECURE IT WITH 4 #8-32 X .25 LG. SCREWS (ITEM #10) AND 4 LOCK WASHERS (ITEM #11).
18. OPEN THE FILTER HSG (ITEM #31) AND INSTALL THE SHIPPING SCREW #8-32 X .28 (ITEM #19) AND A LOCK WASHER (ITEM #11). CLOSE THE FILTER HSG (ITEM #31) AND SECURE THE TWO 1/4 TURN FASTENERS.
19. FILL OUT THE SERIAL LABLE (ITEM #23) BY TYPING AND ATTACH AS SHOWN. THE LEGEND ON SHEET #1 CONTAINS THE INFORMATION FOR EACH VARIATION.





UNIT WEIGHT 44 LB'S.

REVISIONS		
LTR	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
UNIT ASSY BAII-R	E	UA	BAII-R-0	
SCALE NONE	SHEET 2 OF 2	DIST.		

5

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS
PARTS LIST

MADE BY Nelson Roy DATE 3/8/78	CHECKED W. F. McCarthy DATE 16 JUNE 78	SECTION 1
ENG DATE	PROD R. HAMILTON DATE 16 June 78	ISSUED SECT. 1

ITEM NO.	DWG NO. / PART NO.	DESCRIPTION	QUANTITY / VARIATION BA11-RA	BA11-RB
1	C-MD-7418573-0-0	BRACKET, BEZEL MOUNTING	2	2
2	D-AD-7013949-0-0	BEZEL ASSEMBLY WITH SWITCHES	1	1
3	1700083-02	LINE CORD (115V, 60 HZ)	1	-
4	1700090-00	LINE CORD (250V, 50HZ)	-	1
5	D-IA-7013734-0-0	BASIC LOGIC BOX	1	1
6	E-UA-H9273-Ø-Ø	LOGIC ASSEMBLY (H9273)	1	1
7	D-UA-H4Ø3-A-Ø	AC INPUT BOX (H4Ø3-A)	1	1
8	D-UA-H786-Ø-Ø	POWER SUPPLY ASSEMBLY (H786)	1	1
9	C-IA-7418276-0-0	BRACKET, SUPPORT	2	2
10	9006035-01	SCR, PHL HD PAN #8-32 x .25	12	12
11	9006634-00	WASHER, LOCK #8	21	21
12	D-IA-7014093-0-0	HARNESS, AC POWER	1	1
13	C-IA-7014091-0-0	HARNESS, DC	1	1
14	C-IA-7Ø11411-ØK-Ø	CABLE BACKPLANE	1	1
15	E-IA-7418275-0-0	COVER, LOGIC BOX	1	1
16	9008196-00	RECEPTACLE, 1/4 TURN	2	2
17	9006563-00	NUT, KEPS #8 - 32	2	2
18	9007082-00	CABLE CLAMP .31 DIA	1	1
19	9006037-01	SCR, PHL, HD, PAN #8 - 32 x .38	9	9
20	C-IA-7014240-0-0	SPRING, BUTTON ASSEMBLY	1	1
21	9006632-00	WASHER, LOCK INT. TOOTH #4	2	2
22	9008032-01	SCREW, PHL, HD, PAN #4-40 x .19	2	2

TITLE	ASSY NO.	SIZE	CODE	NUMBER	REV.	ECO NO.
UNIT ASSEMBLY BA11-R	E-UA-BA11-R-Ø	A	PL	BA11-R-Ø		
	SHEET 1 OF 2	DIST.				

DEC FORM DEC 16 (325)-1031-N870
DRA 110

MK

DIGITAL EQUIPMENT CORPORATION
MAYNARD, MASSACHUSETTS
PARTS LIST

MADE BY Nelson Roy DATE 3/8/78	CHECKED W. McCarthy DATE 16 JUNE 78	SECTION 1
ENG DATE	PROD R. HAMILTON DATE 16 June 78	ISSUED SECT. 1

ITEM NO.	DWG NO. / PART NO.	DESCRIPTION	QUANTITY / VARIATION BA11-RA	BA11-RB
23	3612450-00	LABEL, SERIAL	1	1
24	9007380-00	TIE, CABLE, SST 1.5M	3	3
25	9007786-00	NUT, TINNERMAN #10 x 32	8	8
26	9007651-00	WASHER, LOCK, EXT TOOTH #10	8	8
27	9006073-03	SCR, PHL, TRUSS HD, #10 - 32 x .50	8	8
28	C-IA-7011411-1A-0	CABLE CONSOLE	1	1
29	9008343-00	FASTENER, 1/4 TURN	2	2
30	9008200-00	RETAINER, 1/4 TURN	2	2
31	D-IA-7013521-0-0	FILTER HOUSING ASSEMBLY	1	1
32	9008404-02	SCREW, PHL, FLAT HD #6-32 x .31	2	2
33	B-MD-7414197-0-0	COVER PLATE	5	5
34	9006010-01	SCR, PHL PAN HD #6-32 x .31 LG	10	10
35	9006633-00	WASHER, INT TOOTH LOCK, #6	10	10
36	3700133-00	PACKAGING INSTRUCTIONS (INTERPLANT)	1	1
37	3700061-00	PACKAGING INSTRUCTIONS (CUSTOMER)	1	1

TITLE	ASSY NO.	SIZE	CODE	NUMBER	REV.	ECO NO.
UNIT ASSEMBLY BA11-R		A	PL	BA11-R-Ø		
	SHEET 2 OF 2	DIST.				

DEC FORM DEC 16 (325) 1031 N870
DRA 110

MK

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION"

NOTES:

NOTES:

1. ~~ADJUST R14 SO THAT THE FREQUENCY MEASURED AT E57 PIN 6 IS 2.45 MHZ MIN TO 2.63 MHZ MAX. THE MODULE MAY BE OPERATED AT OTHER FREQUENCIES FOR TEST PURPOSE ONLY. SEE SHEET (K-9) MANUFACTURING ADJUSTS TO 2.591 MIN TO 2.631 MAX.~~
2. ADJUST R15 SO THAT THE REFRESH FREQUENCY MEASURED AT E78 PIN 8 IS 625 HZ ± 20 HZ. SEE SHEET (K-9).
3. TIME DELAY FROM FALLING EDGE ON E51 PIN 9 TO FALLING EDGE ON E78 PIN 4 SHOULD BE 110 TO 140 NS. R7 MAY BE REMOVED TO INCREASE THE DELAY SLIGHTLY IF LESS THAN 110 NS IS MEASURED. SEE SHEET (K-5).
4. JUMPER FUNCTIONS AND CONFIGURATION ARE AS FOLLOWS:

JUMPER SELECT FEATURES				
JUMPER	SELECTION WHEN INSTALLED	OPTION	SHIPPING STATUS M7264(REV E)	-YA
W1	RESIDENT MEMORY BANK 1	C.O.	R	R
W2	RESIDENT MEMORY BANK 0	C.O.	I	R
W3	DISABLE EVENT LINE	C.O.	P	R
W4	DISABLE MEMORY REFRESH	C.O.	R	I
W5	POWER UP MODE SELECT	C.O.	R	R
W6	POWER UP MODE SELECT	C.O.	R	R
W7	BIASING VOLTAGE VDATA = VROM	F.O.	I	I
W8	BIASING VOLTAGE VCTL = VROM	F.O.	I	I
W9	DISABLE REPLY FROM RESIDENT MEMORY	F.O.	P	I
W10	DISABLE REPLY DURING REFRESH *	C.O.	R	R
W11	ENABLE ON BOARD MEMORY SELECT	C.O.	I	R

* INSTALLED IF RESIDENT RAM IS NOT THE SLOWEST REPLY IN THE SYSTEM.

POWER-UP MODES			
MODE	JUMPERS W6	JUMPERS W5	MODE SELECT
0	R	P	PC AT 24 AND PS AT 26, OR HALT MODE
1	R	I	ODT MICROCODE
2	I	R	PC AT 173000 FOR USER BOOT STRAP
3	I	I	MICRO ADDRESS 3002

I; INSTALLED
R; REMOVED
C.O; CUSTOMER OPTION
F.O; FACTORY OPTION

5. COMPONENTS ARE NOT INSTALLED (RESERVED OPTION) UNLESS SPECIFICALLY INSTRUCTED.
6. DELETED FOR M7264-YA VERSION

~~M7264~~	~~X,Y VBB FOR LSI CHIP SET, -5 VBB FOR MOSTEK 4K RAM~~
~~M7264-YA~~	~~-3.9 VBB FOR LSI CHIP SET, NO ON BOARD RAM~~
~~M7264-BB~~	~~X,Y VBB FOR LSI CHIP SET, -5 VBB FOR INTEL 4K RAM~~
~~M7264-YB~~	~~SAME AS M7264 EXCEPT E58,62 & 70~~
~~M7264-YC~~	~~SAME AS M7264-YA EXCEPT E58,62 & 70~~

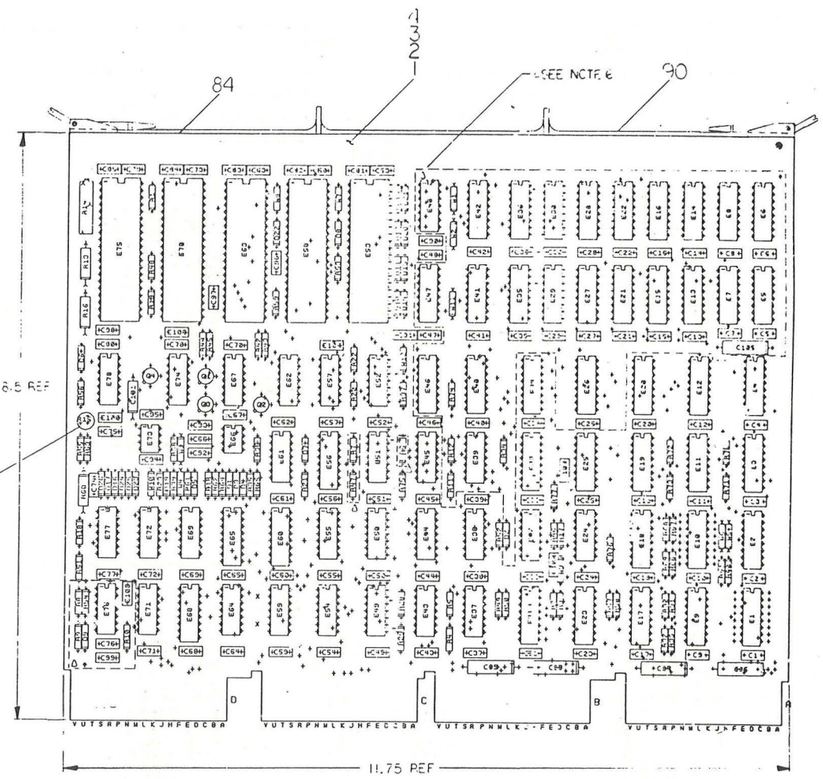
7. SIGNAL DESIGNATION
   XX YYY Z
   - XX: SIGNAL ORIGIN (PAGE) IDENTIFIER
   - YYY: SIGNAL NAME
   - Z: H (HIGH), L (LOW) WHEN ASSERTED
8. ALTERNATE PARTS CHART

ITEM*	PREFERRED PART	ALTERNATE PART
6	1213113-01	7413826-00 AND -01
21,22	1001610-00	1001610-01
9,10	9009149-01	9009149

* THE 23ØØ2B6 IS AN ALLOWABLE SUBSTITUTION FOR THE 23ØØ3B6 ON THE M7264-BB AND M7264-YB. THE 23ØØ3B6 IS PREFERRED.

VARIATION CHART	
VARIATION	DESCRIPTION
M7264-AA	M7264 OR M7264-AB,CB,DB,EB,FB,HB,JB
M7264	PROCESSOR W MOSTEK 4096
M7264-AB	PROCESSOR W INTEL 2104
M7264-BB	PROCESSOR W MOSTEK 4096 AND HYBRID MICROM
M7264-CB	PROCESSOR W FUJITSU 8224
M7264-DB	PROCESSOR W INTEL 2104A
M7264-EB	PROCESSOR W MOSTEK 4027
M7264-FB	PROCESSOR W DEC 4027
M7264-HB	PROCESSOR W MOTOROLA 4027
M7264-JB	PROCESSOR W FUJITSU 8227
M7264-YA	PROCESSOR W NO MOS RAM CS REV W OR LOWER
M7264-YB	PROCESSOR W NO MOS RAM AND HYBRID MICROM
M7264-YC	PROCESSOR W NO MOS RAM CS REV AA OR HIGHER

11. FOR VARIATIONS -BB AND YB ONLY, ADJUST R14 SO THAT THE FREQUENCY MEASURED AT E57, PIN6 IS 2.450 MHZ MIN. TO 2.490 MHZ MAX. FOR ALL OTHER VARIATIONS ADJUST R14 SO THAT THE FREQUENCY MEASURED AT E57, PIN6 IS 2.591 MHZ MIN. TO 2.631 MHZ MAX. SEE SHEET K-9. THE MODULE MAY BE OPERATED AT OTHER FREQUENCIES FOR TEST PURPOSES ONLY.



IC TYPE	GND	+5V	+12V
4K RAM	16	9	8
DEC MHØØ26	3		
DEC 75322	7	14	
DEC 7497	8	16	
DEC 8641	8	16	
DEC 74S257	8	16	
DEC 74S174	8	16	
DEC 74S140	7	14	
DEC 74S139	8	16	
DEC 7475	12	5	
DEC 74S74	7	14	
DEC 7474	7	14	
DEC 7442	8	16	
DEC 74S11	7	14	
DEC 74S1Ø	7	14	
DEC 8837	8	16	
DEC 74S35	7	14	
DEC 74S04	7	14	
DEC 74S2Ø	7	14	
DEC 7400	7	14	

GND AND 5V ARE USUALLY PIN 7 AND 14 RESPECTIVELY EXCEPTIONS ARE STATED ABOVE
IC PIN LOCATIONS

NOTE: FOR PARTS LIST, SEE B-PL-M7264-Ø-Ø

QTY	REF. DESIGNATION	DESCRIPTION	PART NO.	ITEM NO.
		PARTS LIST		
		ETCH BOARD REV. F		

FIRST USED ON OPTION MODEL: KDII-F

DEC NO.	EIA NO.	DEC NO.	EIA NO.

SEMICONDUCTOR CONVERSION CHART

DRN.		DATE
CHK'D.		DATE
		DATE 6-3-76
		DATE 6-3-76
		DATE 6-3-76
NEXT HIGHER ASSY		
SCALE —		
SHEET 1 OF 9		

digital

TITLE: LSI-11 CPU MODULE

SIZE D | CODE CS | NUMBER M7264-0-1 | REV AB | DIST.

DEC FORM NO. DRD 125.B

7

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1972 DIGITAL EQUIPMENT CORPORATION"

PREFIX "K2" APPLIES TO ALL WDAL, WMIB } SIGNALS

K9 -V DATA (SEE NOTE 4) W7 K9 -VROM

C81 .47uf

C84 .47

K9 RPH 3H
K7 RPH 1H
K9 +12VA

CP1611 E53 DATA CHIP

WDAL 00 H … WDAL 15 H

WMIB 00 L … WMIB 15 L

+5V R1 2K R22 2K

K2 MSTB L
K2 MCMD H

+5V R19 2K R18 2K

K2 MCKD L
K2 MCK L

E75 J1

WMIB 00 L … WMIB 15 L

K9 PH2 H
K9 PH4 H

+5V R48 2K

K2 MROM DISABLE L

+5V
K9 RPH 2H
K9 RPH 4 H

C85 .47uf K9 -VROM

WAIT H

K9 -VCTL (SEE NOTE 4) W8 K9 -VROM

C82 .47uf

C83 .47uf

K9 RPH3 H
K9 RPH1 H
K9 +12VA

K8 IOIRQ (1) H
K8 EVIRQ (1) H
K8 IPIRQ (1) H
K8 RFIRQ (1) H
K8 BUSY H
K2 MCOMP H
K8 RESET (1) L
K7 REPLY H
K2 WDOUT H
K2 WWB H
K2 WI9K H
K2 WSYNC H
K2 WDIN H

CP1621B E58 CONTROL CHIP

WMIB 00 L … WMIB 15 L

CP1631 B E63 ROM CHIP 1

WMIB 15 L … WMIB 21 L

K2 MROM DISABLE L

CP1631 B E70 ROM CHIP 2

WMIB 13 L … WMIB 21 L

K2 MROM DISABLE L

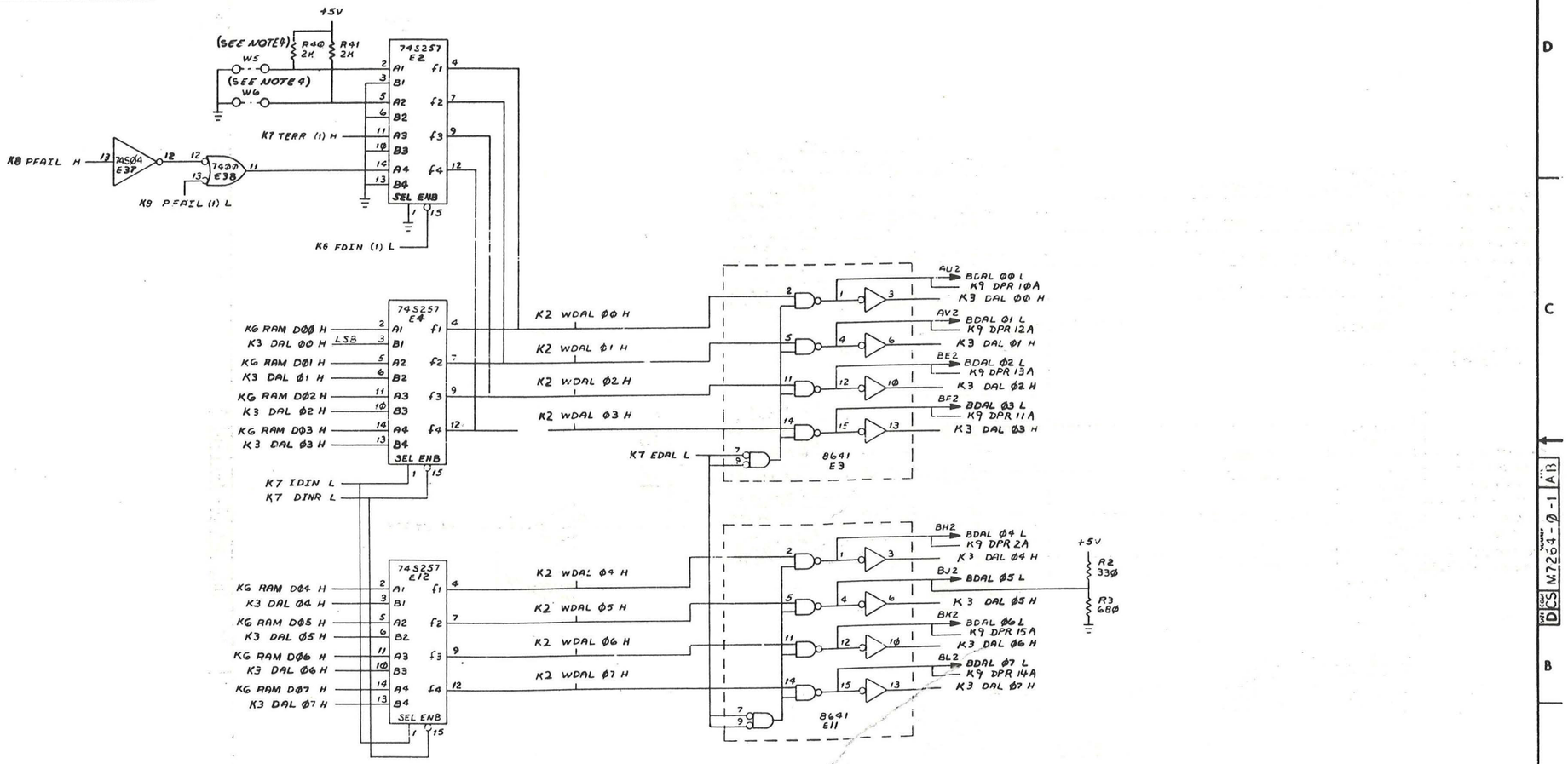
+5V
K9 RPH 2H
K9 RPH 4 H

REVISIONS		
CHK	CHANGE NO	REV

TITLE	SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K2)	D	CS	M7264-0-1	AB
SCALE — SHEET 2 OF 9	DIST.			

DEC FORM NO. DRD 138

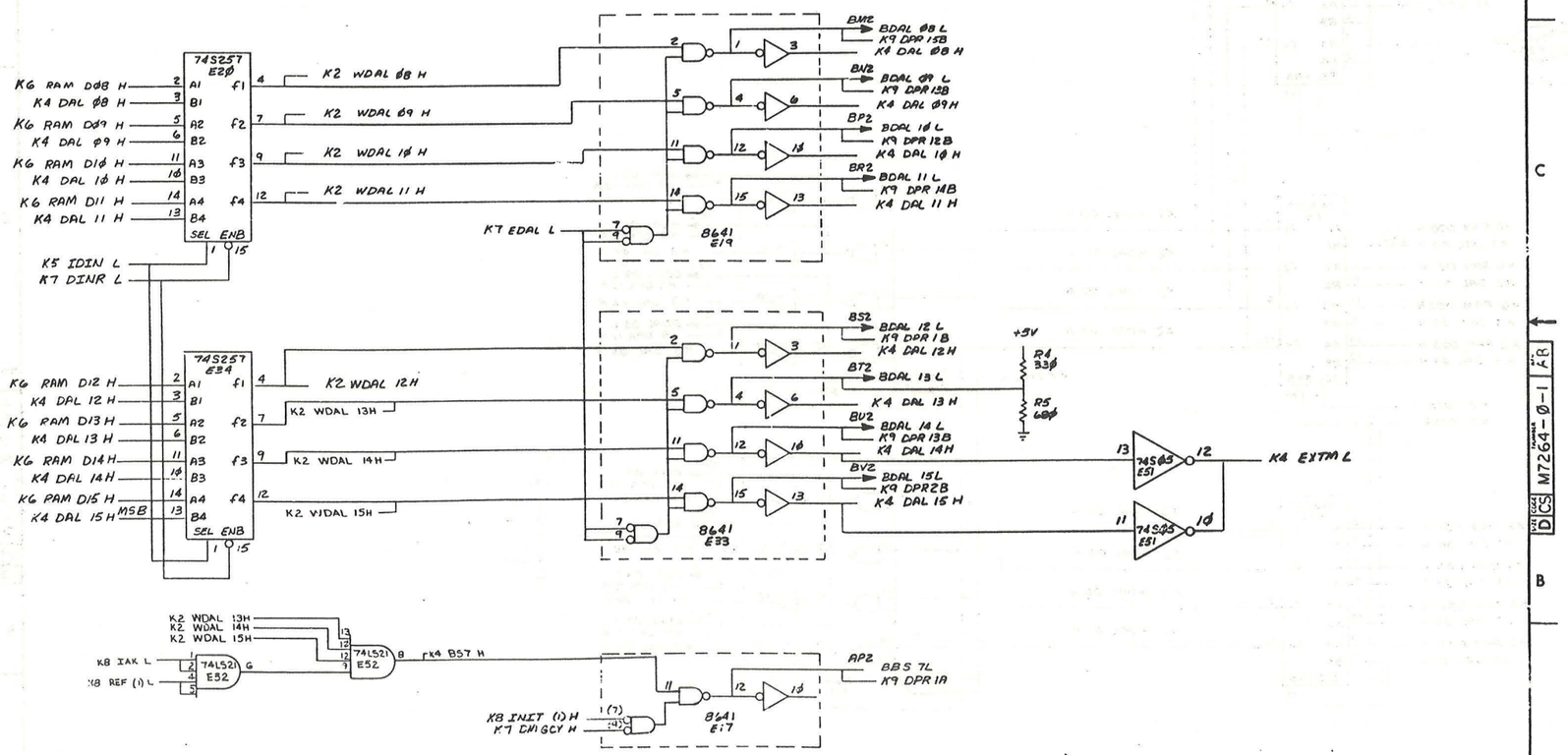
"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1975 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO	REV

TITLE	SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K3)	D	CS	M7264-0-1	AB
SCALE — SHEET 3 OF 9	DIST.			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1975 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K4)	D	CS	M7264-Ø-1	AB
SCALE — / — SHEET 4 OF 9	DIST.			

M7264-Ø-1 AB D CS

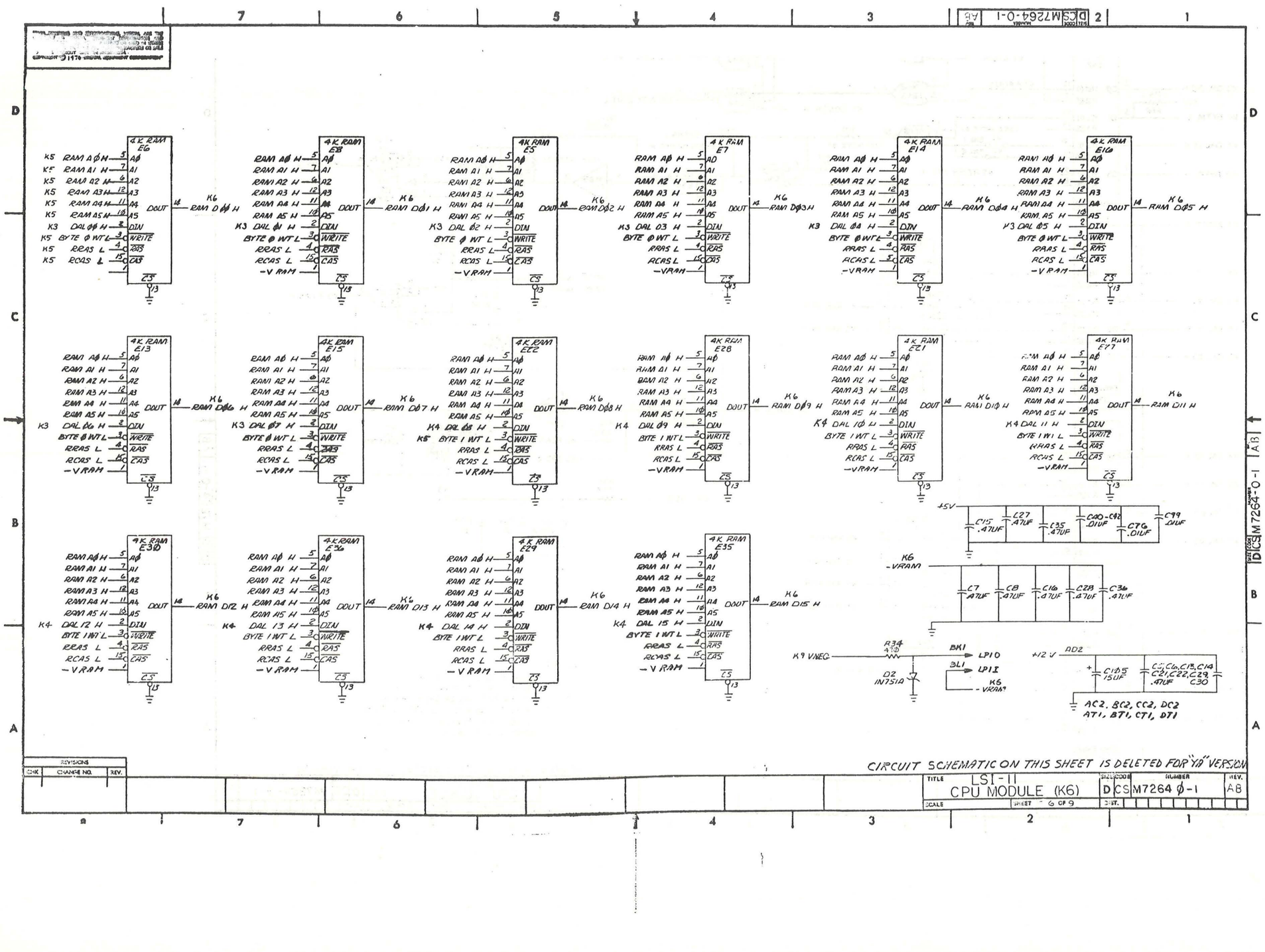
THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION

K5 A Ø(1) L
K7 WTBTR H
K5A Ø(Ø) L
K7 DOUTR H
K5 BYTE 1 WT L
K5 BYTE Ø WT L
K3 DAL ØØ H
+5V
R6 68Ø
K4 EXTM L
K4 DAL 13 H
K8 REFR H
(SEE NOTE 4)
W1 BNK 1
W2 BNK Ø
W11
(SEE NOTE 4)
CF1
K5 SMENB L
K5 BNKOR REF H
K5 REFR (Ø) H
R74 2K
R75 2K
R11 1Ø(22)
K5 RRAS L
(SEE NOTE 3)
R7 6.8K (CEL)
R97 5ØØ (249) 1%
+5V
+3V
K5 AMX H
K5 -5V
R8 133 1%
R9 38.3 1%
R1Ø 133 1%
LP51
LP50
CL1
CK1
C1Ø1 .Ø1Uf
C1Ø3 22Øpf 1%
K5 E CAS H
K7 SYNCR H
R12 1Ø(22)
K5 RCAS L
K5 CAS L
C1Ø2 68Øpf (22Øpf) 1%
R46 5.11K 1% (1.8K 1%)
R17 1.2K (DELETED)
K5 GCAS H
K9 VNEG
R54 27Ø
K5 5V
D9 IN751A 5.1V
C64 .Ø1
C99 .Ø1
K4 DAL 12 H
K3 DAL Ø6 H
K4 DAL 11 H
K3 DAL Ø5 H
K5 RAM A5 H
K5 RAM A4 H
K5 AMX H
K4 DAL 1Ø H
K3 DAL Ø4 H
K4 DAL Ø9 H
K3 DAL Ø3 H
K5 RAM A3 H
K5 RAM A2 H
K5 RAM A1 H
K5 RAM AØ H
K4 DAL Ø8 H
K3 DAL Ø2 H
K3 DAL Ø7 H
K3 DAL Ø1 H
K5 A 4X H
KT SYNCR L

NOTE:
1. COMPONENTS ON THIS SHEET APE DELETED FOR THE YA VAR. WITH THE EXCEPTIONS OF E45, R74 AND R75.
2. NUMBERS IN BRACKETS DENOTE COMPONENT VALUES FOR M7264-EB, FB, HB AND JB VARIATIONS ; 7 PLACES.

CIRCUIT SCHEMATIC ON THIS SHEET IS DELETED FOR "YA" VERSION

TITLE		SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K5)		D	DCS	M7264-Ø-1	A8
SCALE	SHEET 5 OF 9	DIST.			



CIRCUIT SCHEMATIC ON THIS SHEET IS DELETED FOR "YA" VERSION

REVISIONS		
DR	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K6)	D	CS	M7264 0-1	AB
SCALE	SHEET 6 OF 9	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 BY DIGITAL EQUIPMENT CORPORATION"

7497 E65 — +3V; E1, E2, E4, E8, E16, E32; U/C, Z, Y, EN/OUT; CLK STB CLR EN; K7 TERR L; K7 TERR (I) H; K8 INIT (I) L; K8 TF CLR L; K9 PH 3 H; K7 REPLY (I) H

K7 DIN L; K8 DOUT D (I) L; 74S00 E24; 7404 E69; +5V; R52 68Ø; 74S11 E32; K7 REPLY H

K7 RPLYR H; K9 PH 1 H; K8 INIT (I) L; K2 WDOUT H; 74S74 E50; K7 REPLY (I) H; K7 REPLY (I) L

74S10 E55; K7 DOUT (I) H; K7 DOUT (I) L; K9 PH 3 H; K7 SYNC H; 74S74 E50

AE2 BDOUT L; K9 DPR 7B; 8641 E18; K7 DOUTR H

AK2 BWTBT L; K9 DPR 10B; K2 WWB H; K8 INIT (I) H; 8641 E18; K7 WTBTR H

R57 68Ø; R58 33Ø; R67 68Ø; R66 33Ø; R69 68Ø; R68 33Ø; R71 68Ø; R70 33Ø; R73 68Ø; R72 33Ø; BPI; AC1; BAD IG L; AD1; BAD IT L; AA1; ABI; +5V

+5V; R49 33Ø; R50 68Ø; B9ACK L; BN1; 8837 E31; 74S04 E37; K7 SACK R L

K7 DMGCY H; K8 DMG (I) L; 7400 E38; 74S04 E37; K7 DINR H; K5 BNK CR REF H; 74S10 E55; 74S11 E32; K7 EDAL L

K2 WDIN H; 74S04 E37; K7 DIN L; 74S04 E37; K7 DIN H

AH2 BDIN L; K9 DPR 4B; K8 INIT (I) H; 8641 E18; K7 DINR H; FDIN (I) L; 74S20 E24; K7 DINR L

K7 REPLY (I) H; 74S04 E72; TP1; K7 SYNC H; K2 WSYNC H; 74S04 E59; 74S04 E59; K7 SYNC L; K9 PH 3 L; 74S74 E54; K8 IAK L; K7 TSYNC H; K8 INT (I) H; 8641 E18; AJ2 BSYNC L; K9 DPR 6A; K7 SYNC R H; 74S10 E77; K7 SYNCR L

+5V; (SEE NOTE 4); R23 68Ø; W10; K5 REFR (Ø) H; K7 DINR H; K5 GCAS H; 74S10 E62; K7 IDIN L; 74S20 E24; (SEE NOTE 4); W9; K7 DOUTR H; 74S20 E24; K7 TRPLY H; AF2 RRPLY L; K9 DPR 11B; 8641 E23; K7 RPLYR H

REVISIONS		
LTR	CHANGE NO.	REV.

TITLE	LSI-11 CPU MODULE (K7)	SIZE/CODE	NUMBER	REV.
		D	DCS M7264-Ø-1	AB
SCALE	SHEET 7 OF 9	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

AL2 BIRQ L K9 DPR 5A
8837 E31

BR1 BEVNT L K9 DPR 6B
8837 E31

R32 330 R33 330 +5V
AP1 B HALT L

K8 INIT (1) H
K7 DMG CY H
8641 E17
K8 PFAIL H

74S11 E32
R77 330 C107 1200pf

BB1 BPWR OK H K9 DPR 7A
8837 E31

R30 330 R31 680 +5V
ANI BDMR L
8837 E31

K8 DCLD L
K8 PFCLR L
7400 E38
3745204 E37

(SEE NOTE 4) R39 2K +5V
W4
7474 E71
K9 RF OSC L
K8 RF SET L
K8 RF REQ (1) L

(SEE NOTE 4) R38 2K +5V
W3
7474 E71
K8 EF CLR L

74S174 E60
K7 DOUT (1) L
K7 TERR L
CLR CLK
K9 PH 2H

K8 DOUT D (1) L
K8 RFIPQ (1) H
K8 IDIRQ (1) H
K8 EVIRQ (1) H
K8 IPIRQ (1) H
K8 RESET (1) L

BA1 BDC OK H K9 DPR 9A
8837 E31
74S02 E59
K8 DC LO L

+5V R20 680
7400 E72
K8 PFAIL (1) L
7474 E43

K7 SACKR L
7474 E54
K9PH 1 H
K8 INIT (1) L
7400 E72
K8 BUSY H
R51 680 +5V

K9 PH 4 H
7474 E49
K7 SYNC L

K8 DMG (1) H
7474 E49
K8 DMG (1) L

AS2 B DMGO L K9 DPR 9B
8641 E23

K9 PH 3 L
7404 E69
DC1 K8 SPH 3 H

K2 WMIB 21 L
7404 E69
DH1 K8 SROM 3H
K9 PH 3 H
7400 E45

K2 WMIB 20 L
7404 E69
DF1 K8 SROM 2H

K2 WMIB 19 L
7404 E69
DE1 K8 SROM 1H

K2 WMIB 18 L
7404 E69
DD1 K8 SROM ØH

7442 E68
K8 EF CLRL
K8 PF CLRL
K8 TF CLRL

K8 DC LO L
7474 E64
K8 INIT (1) H
K8 INIT (1) L

AT2 BINIT L K9 DPR 3B
8641 E23

CR2 LP2O
CS2 LP2I
CM2 LP3O
CN2 LP3I

K8 IF CLR L

K8 RF SET L
7474 E64
K8 REF (1) L
TP2 K8 T REF (1) H

K8 INIT (1) H
K7 DMG CY H
8641 E17
AR1 B REF L K9 DPR 4A
K8 REF RH

+5V R76 680
8641 E23
AF1 K8 SRUN L
CH1 K8 SRUN L

K8 DC LO L
K8 FDIN (1) H
K8 FDIN (1) H
7474 E44
K7 SYNC H

K2 WIAK H
74S04 E59
K8 IAK L
74S04 E59
K8 IAK (1) H
7474 E43
K9PH 1 L
K7 DIN H

K8 INIT (1) H
K7 DMG CY H
8641 E17
AN2 BIAKO L K9DPR 3A

K9 PH 2H

K8 SPH 3 H
K8 SROM 3H
K8 SROM 2H
K8 SROM 1H
K8 SROM ØH

K8 INIT (1) H
K7 DMG CY H

K8 PFAIL H

K8 BUSY H

K8 DMG (1) H

K8 IAK L

K8 IAK (1) H

K8 INIT (1) H

K8 REF (1) L

K8 FDIN (1) H

K8 DC LO L

K8 INIT (1) L

K8 RF SET L

K8 RF REQ (1) L

K8 RF OSC L

K8 EF CLR L

K8 PF CLR L

K8 TF CLR L

K8 IF CLR L

K8 SRUN L

K8 DMG (1) H

K8 DOUT D (1) L

K8 RFIPQ (1) H

K8 IDIRQ (1) H

K8 EVIRQ (1) H

K8 IPIRQ (1) H

K8 RESET (1) L

K8 DC LO L

K8 PFAIL (1) L

K8 BUSY H

K8 IAK (1) L

K8 FDIN (1) H

K8 INIT (1) L

K8 REF (1) L

K8 DMG (1) L

K8 IAK (1) H

K8 INIT (1) H

K8 REF RH

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

CLK GEN

REF GEN

K9 +12.VA

R16 56 1/2.W

R13 56 1/2 W

C98 .47UF

D6 1N751A 5.1V

C80 .01UF

C100 12.PF

C101 .33UF

(SEE NOTE 1)

R14 5K

R63

R15 2K (SEE NOTE 2)

R55

K9 RF OSC H

K2 MCKD L

K2 MCK L

K2 MSTB L

K9 DIVA(Ø) H

K9 DIVB(Ø) H

C104 68 PF

R21 150

D7 DGG4

K9 PH 1 L

K9 PH 3 L

K9 PH 1 H

K9 PH 2 H

K9 PH 3 H

K9 PH 4 H

K9 +12.VA

R65 33

R35 33

C67 .47

R36 100

K9 DIVB (Ø) H

C93 .47UF

D4 DGG4

D3 DGG4

C92 .47UF

D1 DGG4

+12V AD2

C87 15UF

+5V

R28 330

+3V

R29 680

K9 VNEG

R37 330

R59 *

R53 *

C88 .47UF

K9 -VROM

K9 -VCT L

K9 -VDATA

D5 1N748A 3.9V

D22 *

D8 *

C96 .01UF

C97 .01UF

D26 DGG4

DK1 LP4I

DL1 LP4O

K9 +12.VA

RG4 33

RG2 33

C75 .47

RG1 100Ω

K9 DIVA (Ø) H

C95 .47

D17 DGG4

K9 VDVR

D16 D23 DGG4 D25 DGG4 D24 DGG4

RG0 5Ω 1/2.W

+12. VA

C94 .47 UF

D18 DGG4 D19 DGG4

C74 .47 UF

* SEE NOTE *5 ON SHEET 1

C89 15UF C90 .47UF

AC2,BC2,CC2,DC2 AT1,BT1,CT1,DT1

C53 .47UF C58 .47UF C63 .47UF C73 .47UF C79 .47UF

C1-C4 .01

C9-C12 .01

C17-C20 .01

C23-C26 .01

C31-C34 .01

C37-C52 .01

C54-C57 .01

C59-C62 .01

C86 15UF C88 1.UF

AC2,BC2,CC2,DC2 AT1,BT1,CT1,DT1

C70 .47UF

C77 .01

C76 .01

C72 .01

C71 .01

C69 .01

C68 .01

C65 .01

R48 22Ω

R43 22Ω

R45* 22Ω

R44 27Ω

D20 DGG4 D21 DGG4 DK0 DGG4 D11 DGG4 D14 DGG4 D15 DGG4 D12 DGG4 D13 DGG4

K9 RPH 1 H K9 RPH 2 H K9 RPH 4 H K9 RPH 3 H

R56 2K

Q1 Q2 Q3 Q4

C78 .47UF

* FOR VARIATIONS -YB AND -ØB ONLY, R45 IS 15Ω

K9 DPR 15A 14A 13A 12A 11A ØA 9A 7A 6A 5A 4A 3A 2A 1A

K9 DPR 15B 14B 13B 12B 11B ØB 9B 7B 6B 5B 4B 3B 2B 1B

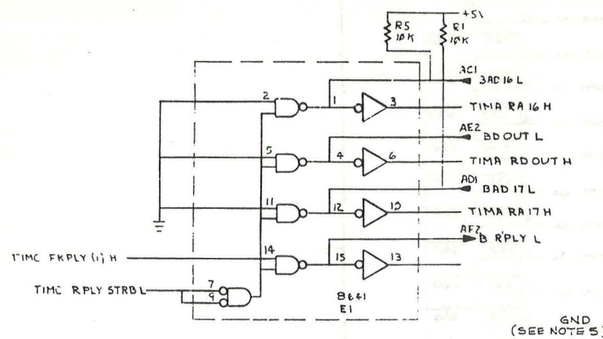
330 680

+5V

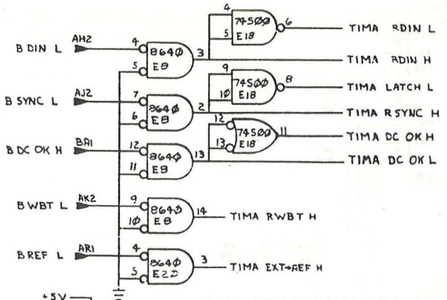
REVISIONS		
CHK	CHANGE NO.	REV.

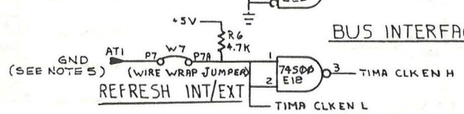
TITLE	SIZE	CODE	NUMBER	REV.
LSI-11 CPU MODULE (K9)	D	DCS	M7264-Ø-1	A8
SCALE	SHEET 9 OF 9	DIST		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976, DIGITAL EQUIPMENT CORPORATION"

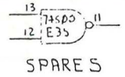


BUS INTERFACE LOGIC



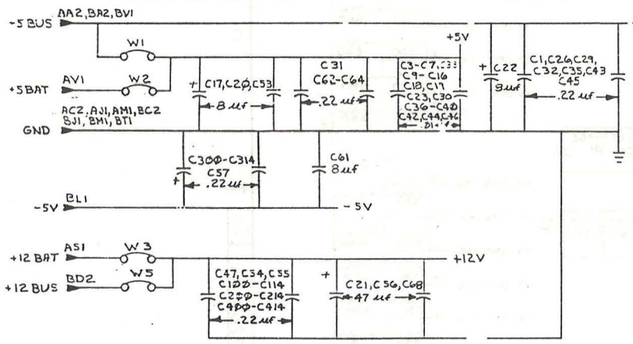


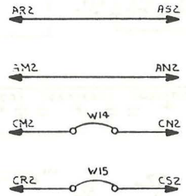
REFRESH INT/EXT



SPARES

BUS INTERFACE LOGIC





BUS GRANT CONTINUITY JUMPERS

W6	W7	REFRESH OPTIONS
IN	IN	INTERNAL REFRESH WITH NO B R'PLY
IN	OUT	EXTERNAL REFRESH WITH NO B R'PLY
OUT	IN	ILLEGAL
OUT	OUT	EXTERNAL REFRESH WITH B R'PLY

REFRESH SELECTION

M7955 VARIATIONS

JUMPERS	-AA	-AB	-AC	-AD
W4	1	1	1	1
W8	Ø	1	1	1
W12	Ø	Ø	1	1
W16	Ø	Ø	Ø	1

ALL JUMPERS NOTED AS Ø MUST NOT BE INSTALLED

STARTING ADDRESS	SWITCH S1 SETTINGS -1	-2	-3	-4	-5
Ø	1	1	1	1	1
4K	Ø	1	1	1	1
8K	1	Ø	1	1	1
12K	Ø	Ø	1	1	1
16K	1	1	Ø	1	1
20K	Ø	1	Ø	1	1
24K	1	Ø	Ø	1	1
28K	Ø	Ø	Ø	1	1
32K	1	1	1	Ø	1
36K	Ø	1	1	Ø	1
40K	1	Ø	1	Ø	1
44K	Ø	Ø	1	Ø	1
48K	1	1	Ø	Ø	1
52K	Ø	1	Ø	Ø	1
56K	1	Ø	Ø	Ø	1
60K	Ø	Ø	Ø	Ø	1
64K	1	1	1	1	Ø
68K	Ø	1	1	1	Ø
72K	1	Ø	1	1	Ø
76K	Ø	Ø	1	1	Ø
80K	1	1	Ø	1	Ø
84K	Ø	1	Ø	1	Ø
88K	1	Ø	Ø	1	Ø
92K	Ø	Ø	Ø	1	Ø
96K	1	1	1	Ø	Ø
100K	Ø	1	1	Ø	Ø
104K	1	Ø	1	Ø	Ø
108K	Ø	Ø	1	Ø	Ø
112K	1	1	Ø	Ø	Ø
116K	Ø	1	Ø	Ø	Ø
120K	1	Ø	Ø	Ø	Ø
124K	Ø	Ø	Ø	Ø	Ø

OVER LAPPING IS NOT ALLOWED IN ANY PERIPHERAL PAGE

1 = ON
Ø = OFF

EXCEPT M7955-AD
EXCEPT M7955-AC & AD
EXCEPT M7955-AB, AC, & AD

STARTING ADDRESS SELECTION

IC PIN LOCATIONS

IC TYPE	GND	+5	+12
555 @ E2	1	8	
555 @ E44	1		8
7493	10	5	
7483A	12	5	
74LS75	12	5	
8640	1	8	
OTHER 16 PIN I.C.'S	8	16	
OTHER 14 PIN I.C.'S	7	14	

NOTES:

1. OPTION DESIGNATION:
   M7955-AA = M7955-YA OR M7955-YE
   M7955-AB = M7955-YB OR M7955-YF
   M7955-AC = M7955-YC OR M7955-YH
   M7955-AD = M7955-YD OR M7955 YJ
2. ALL 8641 BUS REC/DR PLUS 7483A ADDER AND 74S138 DECODER ARE POWERED BY +5 BUS. ALL OTHER CIRCUITS GO TO +5V AS SELECTED BY W1 & W2.
3. W1, W2, W3, W5, W14, W15 ARE MACHINE INSERTABLE JUMPERS. W4, W6, W7, W8, W12, W16 ARE WIRE WRAP JUMPERS.
4. ALL BDAL-XX CONNECTIONS ARE BI-DIRECTIONAL LINES
5. DURING MANUFACTURING TEST GROUND MAY BE REMOVED

NOT USED REF DES

C100-C114, C200-C214, W3, W12, W16
C100-C114, C200-C214, W12, W16
C100-C114, W16

REVISIONS			
SYM	CHANGE NO.		REV.
NZ	M7955-0000414		C
	J. CARTER 10-28-77		
	J. STEGEMAN		
	M7955-0000?		D
	P. DURANT		

ETCH REV	D-P2		
DRN.			
CHK'D		11/21/76	
ENG.			
PROJ. ENG.			
MFGD.			
NEXT HIGHER ASSY.			
3-000-M7955-0			
SCALE			
SHEET 1 OF 3			

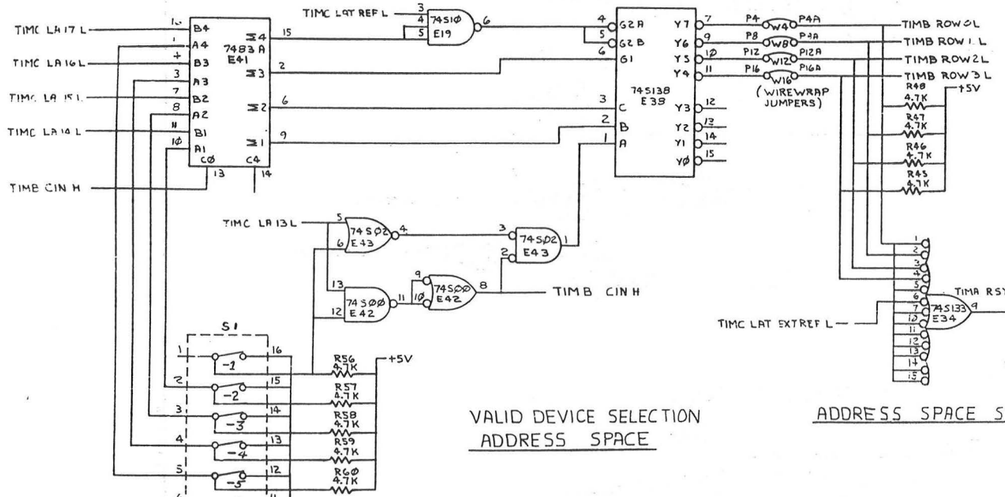
FIRST USED ON MSVII-C digital

TITLE MSVII-C MOS MEMORY, LSI (TIMA)

SIZE	CODE	NUMBER	REV.
D	CS	M7955-0-1	D
DIST.			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1979 DIGITAL EQUIPMENT CORPORATION"

ROW SELECTION



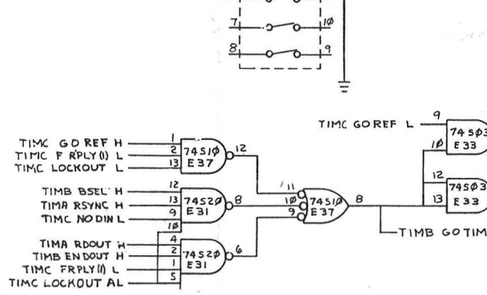
FAILING ROW	MEMORY SIZE			
	M7955-AA(4K)	M7955-AB(8K)	M7955-AC(12K)	M7955-AD(16K)
ROW Ø	NO SWAPPING POSSIBLE	P4 TO P8A	REMOVE W4 ADD JUMPER P4 TO P12A	REMOVE W4 ADD JUMPER P4 TO P16A
ROW 1		* REMOVE W8	REMOVE W8 ADD JUMPER P8 TO P12A	REMOVE W8 ADD JUMPER P8 TO P16A
ROW 2			* REMOVE W12	REMOVE W12 ADD JUMPER P12 TO P16A
ROW 3				* REMOVE W16

1. DETERMINE MEMORY SIZE & REMOVE UPPER 4K ROW JUMPER. (INDICATED BY *)
2. LOCATE FAILING ROW, REMOVE SPECIFIED ROW JUMPER, ADD SWAPPING JUMPER. IF FAILING ROW IS UPPER ADDRESS SPACE ROW, THERE IS NO SWAPPING JUMPER
3. THIS RECONFIGURED MEMORY BLOCK STILL HAS THE SAME STARTING ADDRESS AS BEFORE. IT OCCUPIES 4K LESS ADDRESS SPACE THAN BEFORE. RECONFIGURE STARTING ADDRESSES OF MEMORIES ABOVE THIS ONE IF NECESSARY SO THERE WILL BE CONTISUOUS ADDRESS SPACE.

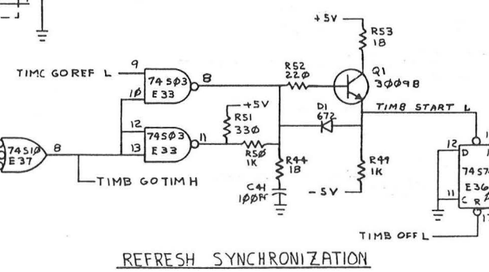
ROW SWAPPING MATRIX

VALID DEVICE SELECTION ADDRESS SPACE

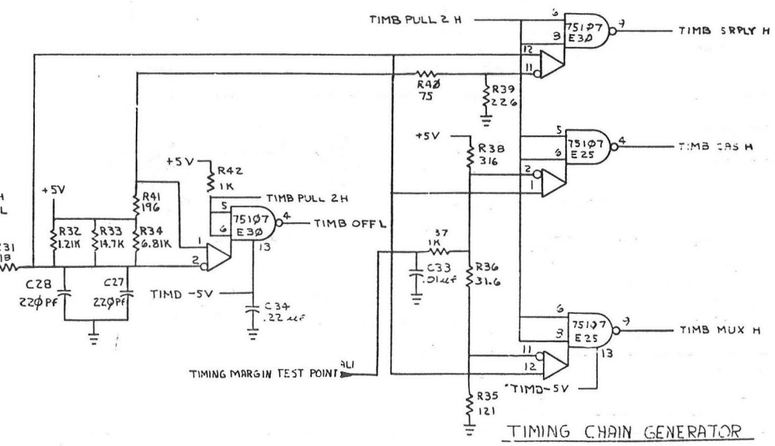
ADDRESS SPACE SIZE & MAPPING



MEMORY CYCLE TYPE



REFRESH SYNCHRONIZATION



TIMING CHAIN GENERATOR

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE		SIZE	CODE	NUMBER	REV.
M5VII-C MOS MEMORY, LSI	(TIM3)	D	CS	M7955-Ø-1	D
SCALE	SHEET 2 OF 3	DIST.			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION"

TIMD DI Ø0 H
TIMD DI Ø1 H
TIMD DI Ø2 H
TIMD DI Ø3 H
TIMA LATCH L
74LS75 E22
TIMC LA Ø0 H
TIMC LA Ø0 L
TIMC LA Ø1 H
TIMC LA Ø2 H
TIMC LA Ø3 H

TIMD DI Ø4 H
TIMD DI Ø5 H
TIMD DI Ø6 H
TIMD DI Ø7 H
74LS75 E24
TIMC LA Ø4 H
TIMC LA Ø5 H
TIMC LA Ø6 H
TIMC LA Ø7 H

TIMD DI Ø8 H
TIMD DI Ø9 H
TIMD DI 10 H
TIMD DI 11 H
74LS75 E27
TIMC LA Ø8 H
TIMC LA Ø9 H
TIMC LA 10 H
TIMC LA 11 H

TIMD DI 12 H
TIMD DI 13 H
TIMD DI 14 H
TIMD DI 15 H
74LS75 E29
TIMC LA 12H
TIMC LA 13L
TIMC LA 14L
TIMC LA 15L

TIMA RA 16 H
TIMA RA 17 H
TIMA RWBT H
TIMA EXT REF H
74LS75 E15
TIMC LA 16 L
TIMC LA 17L
TIMC LWBT H
TIMC LAT REF L

ADDRESS+ CONTROL LATCHES

TIMD RA Ø H
TIMC LA Ø7 H
TIMC LA Ø1 H
74S153 E3
TIMC MA Ø H

TIMD RA 1 H
TIMC LA Ø8 H
TIMC LA Ø2 H
TIMC MA 1 H

TIMD RA 2 H
TIMC LA Ø9 H
TIMC LA Ø3 H
74S153 E10
TIMC MA 2 H

TIMD RA 3H
TIMC LA 10 H
TIMC LA Ø4 H
TIMC MA 3 H

TIMD RA 4 H
TIMC LA 11 H
TIMC LA Ø5 H
74S153 E17
TIMC MA 4H

TIMD RA 5 H
TIMC LA 12 H
TIMC LA Ø6 H
TIMC MA 5H

TIMC INTREF(1) H
TIMB MUX H

ADDRESS MUX'S

TIMB PULL 2H
TIMB TRAS(1) L
74S74 E36
TIMC RESET L
74S03 E33
+5V
R55 2.61K
R54 10.7K
C24 220 PF
B640 E27
TIMB OFF L
TIMB TRAS (1) L
74S12 E19
TIMC LOCKOUT L
TIMA DC CK L
B640 E27
TIMC LOCKOUT AL

BUS CYCLE HOLD OFF

TIMC INTREF 1) H
TIMC LAT EXT REF H
74S02 E39
P6 W6 56A
TIMC REF L
(WIRE WRAP JUMPER)
+5V
R29 4.7 K
74S10 E37
TIMB SRPLY H
B640 E29
R30 18
C25 100 PF
74S74 E49
TIMC FRPLY (1) H
TIMC FRPLY(1) L
TIMA RDOUT H
TIMA RDIN H
74S02 E43
TIMC RPLY STRB L
TIMA RSYNC H

REFRESH RPLY SELECT

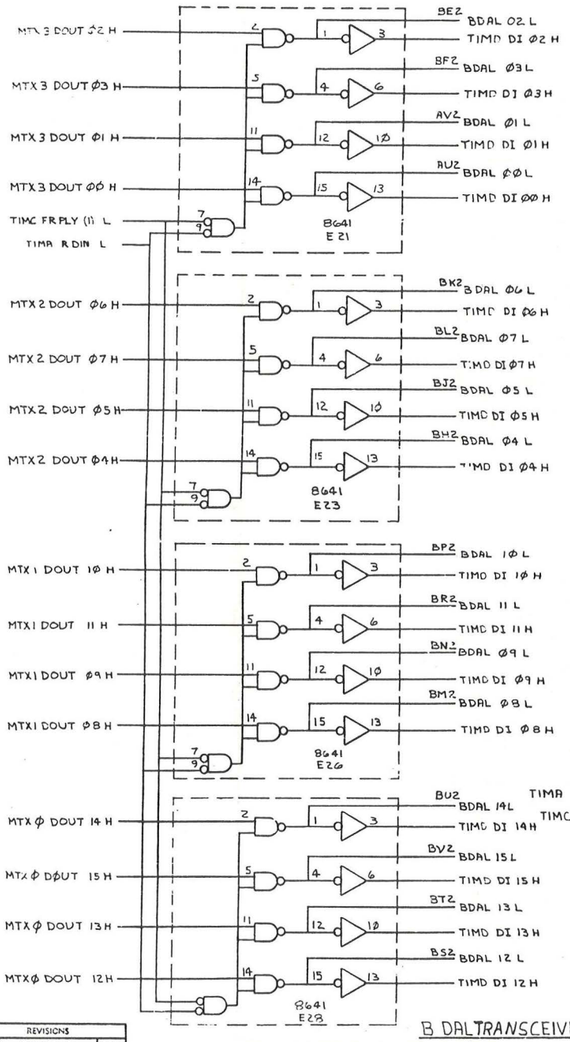
TIMA CLK EN H
TIMD REF CLK H
74S74 E32
TIMC PULL 1H
R43 1K
74S08 E35
TIMC GO REF L
TIMC GO REF H
TIMB GOTIM H
74S74 E40
TIMB OFF L
TIMC INTREF(1) H
74S08 E35
TIMB SRPLY H
TIMB TRAS(1) L
74S02 E39
74S32 E31
TIMC PULL 1 H
TIMC LWBT H
74S02 E43
TIMC NODIN L
74S74 E32
TIMA RSYNC H

TIMA CLK EN L
TIMC LAT EXT REF H
74S08 E35
TIMC LAT EXT REF L

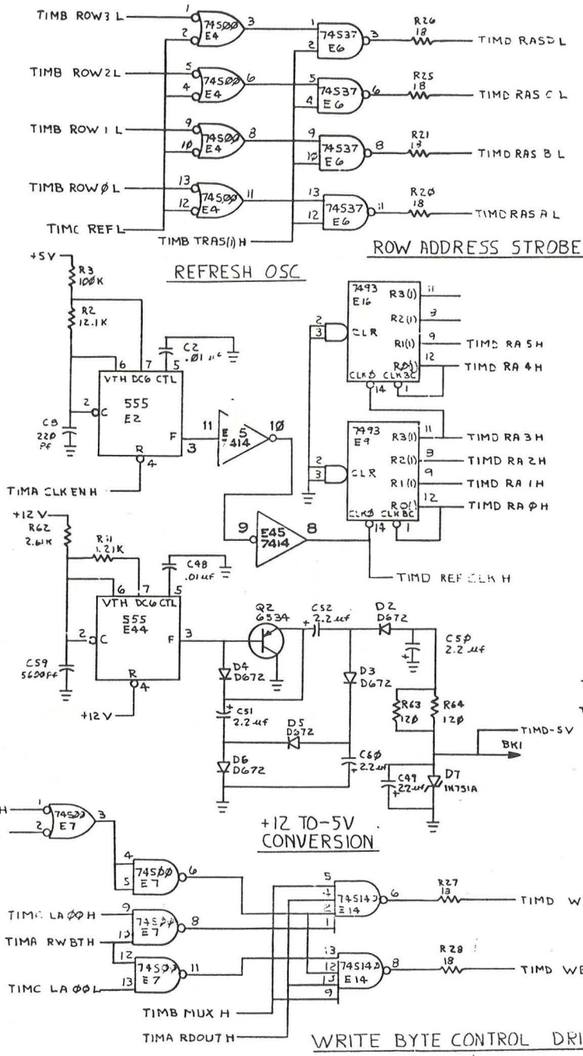
REVISIONS		
CHK	CHANGE NO.	REV.

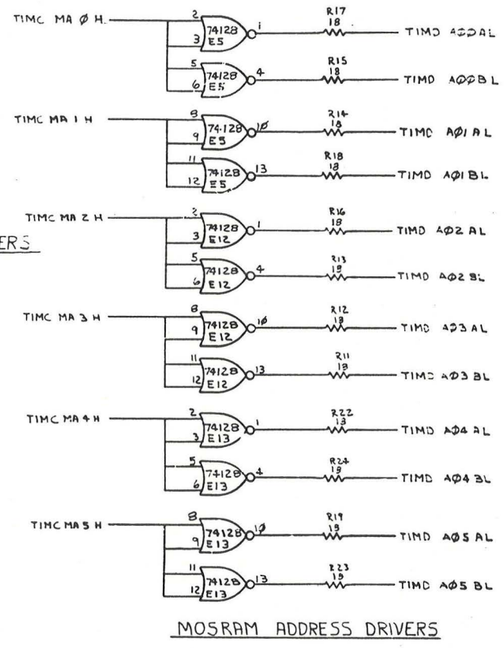
TITLE	SIZE	CODE	NUMBER	REV.
MSVII-C MOS MEMORY, LSI (TIMC)	D	CS	M7955-0-1	D
SCALE	SHEET 3 OF 3	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"



B DALTRANSCEIVERS



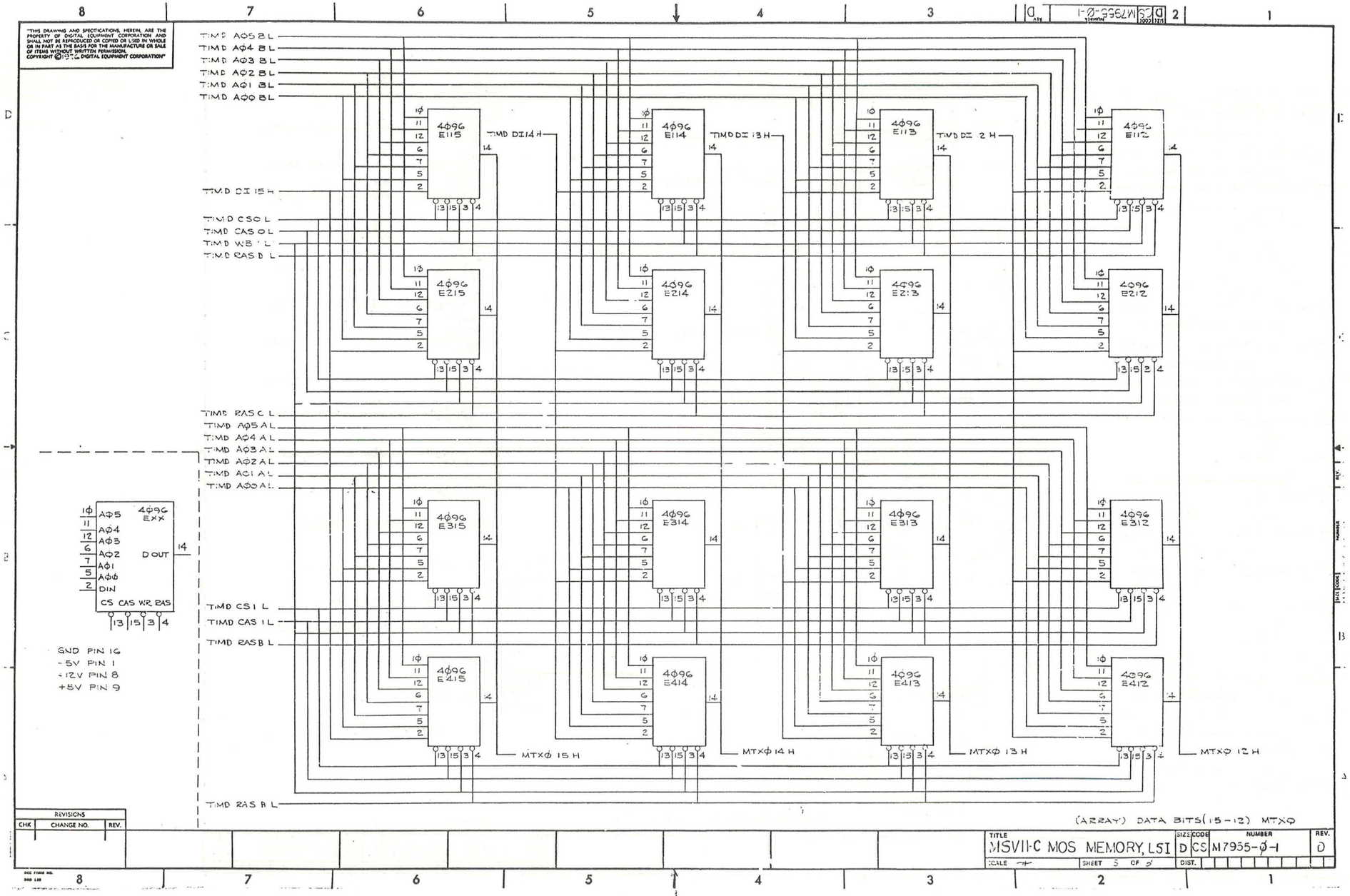


REVISIONS		
CHK	CHANGE NO.	REV.

(TIMD)

TITLE	SIZE	CODE	NUMBER	REV.
MSV11-C MOS MEMORY, LSI (TIMD)	D	CS	M7955-Ø-1	D
SCALE —	SHEET 4 OF 8	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
MSVII-C MOS MEMORY, LSI	D	CS	M7955-Ø-1	D
SCALE —	SHEET 5 OF 5	DIST.		

20

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1978 DIGITAL EQUIPMENT CORPORATION"

TIMD A$5 BL
TIMD A$4 BL
TIMD A$3 BL
TIMD A$2 BL
TIMD A$1 BL
TIMD A$$ BL

TIMD DI 11 H
TIMD CS$ L
TIMD CAS $L
TIMD WB I L
TIMD RAS D L

TIMD RAS C L
TIMD A$5 A L
TIMD A$4 A L
TIMD A$3 A L
TIMD A$2 A L
TIMD A$1 A L
TIMD A$$ A L

TIMD CS I L
TIMD CAS I L
TIMD RAS B L

TIMD RAS A L

4$96 E111 — TIMD DI $H
4$96 E11$ — TIMD DI $9 H
4$96 E1$9 — TIMD DI $8 H
4$96 E1$8
4$96 E211
4$96 E21$
4$96 E2$9
4$96 E2$8
4$96 E311
4$96 E31$
4$96 E3$9
4$96 E3$8
4$96 E411 — MTX1 11 H
4$96 E41$ — MTX1 1$ H
4$96 E4$9 — MTX$ $9 H
4$96 E4$8 — MTX1 $8 H

4$96 EXX
A$5 A$4 A$3 A$2 A$1 A$$ DIN
DOUT
CS CAS WR RAS

GND PIN 16
-5V PIN 1
+12V PIN 8
+5V PIN 9

(ARRAY) DATA BITS (11- 8) MTX1

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
1K STATIC MOS MEMORY, LSI	D	CS	M7955-0-1	D
SCALE —	SHEET 1 OF 3		DWG.	

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION"

TIMD AΦ5 BL
TIMD AΦ4 BL
TIMD AΦ3 BL
TIMD AΦ2 BL
TIMD AΦ1 BL
TIMD AΦΦ BL

TIMD DIΦ7H
TIMD CSΦ L
TIMD CASΦL
TIMD WBΦL
TIMD RAS D L

TIMD RAS C L
TIMD AΦ5 AL
TIMD AΦ4 AL
TIMD AΦ3 AL
TIMD AΦ2 AL
TIMD AΦ1 AL
TIMD AΦΦ AL

TIMDCS 1L
TIMD CAS 1 L
TIMD RAS BL

TIMD RAS A L

4ΦΦ6 EIΦ7 — TIMDDIΦGH
4ΦΦ6 EIΦ6 — TIMDDIΦ5H
4ΦΦ6 EIΦ5 — TIMDDIΦ4H
4ΦΦ6 EIΦ4
4ΦΦ6 E2Φ7
4ΦΦ6 E2Φ6
4ΦΦ6 E2Φ5
4ΦΦ6 E2Φ4
4ΦΦ6 E3Φ7
4ΦΦ6 E3Φ6
4ΦΦ6 E3Φ5
4ΦΦ6 E3Φ4
4ΦΦ6 E4Φ7 — MTX2 Φ7 H
4ΦΦ6 E4Φ6 — MTX2 Φ6 H
4ΦΦ6 E4Φ5 — MTX2 Φ5 H
4ΦΦ6 E4Φ4 — MTX2 Φ4 H

4ΦΦ6 EXX
AΦ5 AΦ4 AΦ3 AΦ2 AΦ1 AΦΦ DIN CS CAS WE RAS — DOUT

GND PIN 16
-5V PIN 1
+12V PIN 8
+5V PIN 9

(ARRAY) DATA BITS(7-4) MTX2

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
M5YII-C MOS MEMORY, LSI	D	CS	M7955-Ø-1	D
SCALE	SHEET 7 OF 9	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1976 DIGITAL EQUIPMENT CORPORATION"

TIMD AØ5BL
TIMD AØ4BL
TIMD AØ3BL
TIMD AØ2BL
TIMD AØ1BL
TIMD AØØBL

TIMD DIØ3H
TIMD CSØL
TIMD CASØL
TIMD WBØL
TIMD RASØL

TIMD EASCL
TIMD AØ5AL
TIMD AØ4AL
TIMD AØ3AL
TIMD AØ2AL
TIMD AØ1AL
TIMD AØØAL

TIMD CS1L
TIMD CAS1L
TIMD RASBL

TIMD EASAL

4Ø96 EXX: 1Ø A5, 11 A4, 12 A3, 6 A2, 7 A1, 5 AØ, 2 DIN, CS CAS WE RAS, 13 15 3 4, DOUT 14

GND PIN 16
-5V PIN 1
+12V PIN 8
+5V PIN 9

4Ø96 E1Ø3 — TIMD DIØ3H — MTX3 Ø3 H
4Ø96 E1Ø2 — TIMD DIØ1H — MTX3 Ø2 H
4Ø96 E1Ø1 — TIMD DIØØH — MTX3 Ø1 H
4Ø96 E1ØØ — MTX3 ØØ H
4Ø96 E2Ø3
4Ø96 E2Ø2
4Ø96 E2Ø1
4Ø96 E2ØØ
4Ø96 E3Ø3
4Ø96 E3Ø2
4Ø96 E3Ø1
4Ø96 E3ØØ
4Ø96 E4Ø3
4Ø96 E4Ø2
4Ø96 E4Ø1
4Ø96 E4ØØ

(ARRAY) DATA BITS (Ø3-ØØ) MTX3

REVISIONS			
CHK	CHANGE NO.	REV.	

TITLE: MSVII-C MOS MEMORY, LSI
SIZE D CODE CS NUMBER M7955-Ø-1 REV D
SCALE — SHEET 3 OF 3 DIST.

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977, DIGITAL EQUIPMENT CORPORATION"

NOTES:

1. ● = WIRE WRAP PIN

2.

TYPE MEMORY	JUMPER CONFIGURATION
NON PARITY	5 TO 7
PARITY	5 TO 6

3.

TYPE	JUMPER CONFIGURATION
4K	17 TO 15 & 17 TO 14
8K	17 TO 15 & 12 TO 14
16K	16 TO 15 & 16 TO 14
32K	16 TO 15 & 10 TO 14

4.

STARTING ADDRESS		SWITCH SETTING (X = OFF)				
DEC (K)	OCTAL	S1-1	S1-2	S1-3	S1-4	S1-5
0	000000					
4	020000					X
8	040000				X	
12	060000				X	X
16	100000			X		
20	120000			X		X
24	140000			X	X	
28	160000			X	X	X
32	200000		X			
36	220000		X			X
40	240000		X		X	
44	260000		X		X	X
48	300000		X	X		
52	320000		X	X		X
56	340000		X	X	X	
60	360000		X	X	X	X
64	400000	X				
68	420000	X				X
72	440000	X			X	
76	460000	X			X	X
80	500000	X		X		
84	520000	X		X		X
88	540000	X		X	X	
92	560000	X		X	X	X
96	600000	X	X			
100	620000	X	X			X
104	640000	X	X		X	
108	660000	X	X		X	X
112	700000	X	X	X		
116	720000	X	X	X		X
120	740000	X	X	X	X	
124	760000	X	X	X	X	X

5. NORMAL JUMPER 1 TO 3.
OPTIONAL JUMPER 1 TO 2
WILL ALLOW USER TO EXTEND USABLE
MEMORY SPACE FROM 28K TO 30K.

6.

JUMPER	BATTERY BACK-UP	NON BATTERY BACK-UP
W2	OUT	IN
W3	OUT	IN
W4	IN	OUT
W5	IN	OUT

7.

MEMORY SIZE	W1	R3
4K	IN	OUT
8K	IN	OUT
16K	OUT	IN
32K	OUT	IN

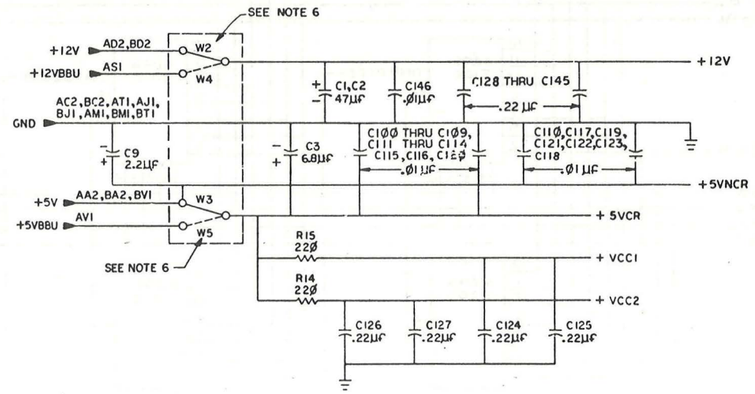
8.

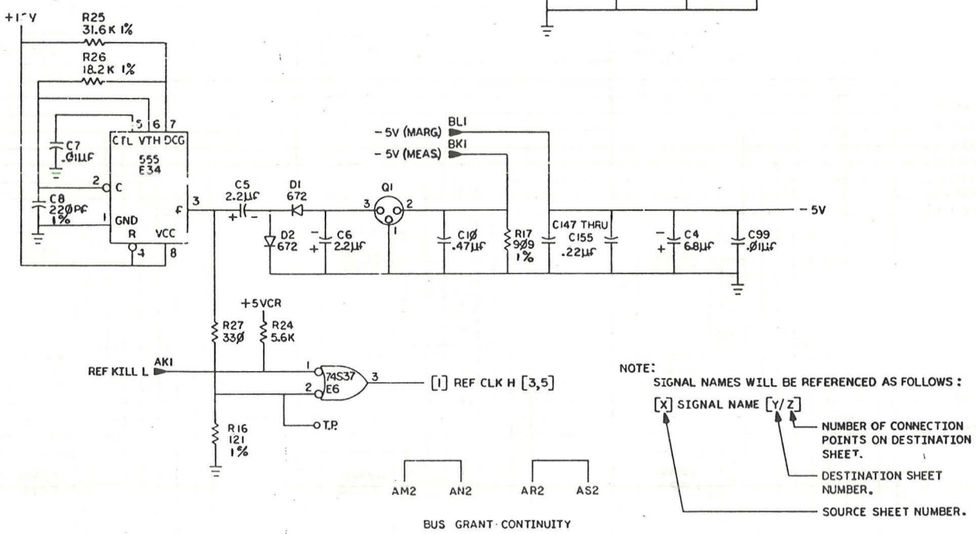
OPTION DESIGNATION	PRIMARY VARIATION	SYSTEM DESCRIPTION
MSV11-CA	M8044-AA	4K X 16 BITS
MSV11-DB	M8044-BA	8K X 16 BITS
MSV11-DC	M8044-CA	16K X 16 BITS
MSV11-DD	M8044-DA	32K X 16 BITS

9.

IC TYPE	+5V	GND	+12V	-5V
555		1	8	
MOS	9	16	8	1
74S373	20	10		
74LS393	14	7		
74S280	14	7		
74S74	14	7		
74S10	14	7		
74S11	14	7		
8641	16	8		
8640	8	1		
74S240	20	10		
74S37	14	7		
74S22	14	7		
1K×4 PROM	18	9		
74S02	14	7		
74LS75	5	12		
74S32	14	7		
74LS10	14	7		
74LS240	20	10		

10. THESE I.C.'S USED ONLY IN 8K & 32K CONFIG'S.





NOTE:
SIGNAL NAMES WILL BE REFERENCED AS FOLLOWS:
[X] SIGNAL NAME [Y/Z]
- NUMBER OF CONNECTION POINTS ON DESTINATION SHEET.
- DESTINATION SHEET NUMBER.
- SOURCE SHEET NUMBER.

REVISIONS			
DIR	CHANGE NO.	REV.	
CJK	M8044-CCCO1	B	
STEVE MILLER 1 MAY 78			
R GIVEN			
R. GIVEN 22-2-78			

DRN Stan Garcia	10-25-77	FIRST USED ON	digital
CHK'D R. GIVEN	2-21-78	MSV11-D	
ENG. R GIVEN	2-21-78	TITLE	
PROJ. ENG. R. Given	2-21-78	32K 16BIT MOS MEM	
PROD. R. Given	2-21-78		
NEXT HIGHER ASSY.			
B-DD-M8044-0		SIZE D	CODE CS
SCALE		NUMBER M8044-0-1	REV. B
SHEET 1 OF 7		DIST.	

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

BDCOK H — BA1 — 8640 E22 — [2] DCOK L [4]
BREF L — AR1 — 8640 E22 — [2] EXT REF H [4]

BDAL12 L — BS2 — SEE NOTE 5

BBS7 L — AP2 — 8640 E22 — [2] BANK SEL 7 H [4]
[2] WTBT H [4]
BWTBT L — AK2 — 8640 E22 — 74S240 E18 (1) — [2] WTBT L [4,5]

BDAL L — AU2 — [3] LAT DØØ H [2] — [2] DAL ØØ H [4,6]
BDAL1 L — AV2 — [3] LAT DØ1 H [2] — [2] DAL Ø1 H [3,6]
BDAL2 L — BE2 — [3] LAT DØ2 H [2] — [2] DAL Ø2 H [3,6]
BDAL3 L — BF2 — [3] LAT DØ3 H [2] — [2] DAL Ø3 H [3,6]
[5] RPLY L [2/5] — [2] DIN L [2/5] — 8641 E51

BDAL4 L — BH2 — [3] LAT DØ4 H [2] — [2] DAL Ø4 H [3,6]
BDAL5 L — BJ2 — [3] LAT DØ5 H [2] — [2] DAL Ø5 H [3,6]
BDAL6 L — BK2 — [3] LAT DØ6 H [2] — [2] DAL Ø6 H [3,6]
BDAL7 L — BL2 — [3] LAT DØ7 H [2] — [2] DAL Ø7 H [3,6]
[5] RPLY L [2/5] — [2] DIN L [2/5] — 8641 E57

BDAL8 L — BM2 — [3] LAT DØ8 H [2] — [2] DAL Ø8 H [3,7]
BDAL9 L — BN2 — [3] LAT DØ9 H [2] — [2] DAL Ø9 H [3,7]
BDAL10 L — BP2 — [3] LAT D10 H [2] — [2] DAL 10 H [3,7]
BDAL11 L — BR2 — [3] LAT D11 H [2] — [2] DAL 11 H [3,7]
[5] RPLY L [2/5] — [2] DIN L [2/5] — 8641 E63

BDAL12 L — BS2 — [3] LAT D12 H [2] — [2] DAL 12 H [3,7]
BDAL13 L — BT2 — [3] LAT D13 H [2] — [2] DAL 13 H [3,4,7]
BDAL14 L — BU2 — [3] LAT D14 H [2] — [2] DAL 14 H [3,4,7]
BDAL15 L — BV2 — [3] LAT D15 H [2] — [2] DAL 15 H [4,7]
[5] RPLY L [2/5] — [2] DIN L [2/5] — 8641 E69

BAD 16 L — AC1 — [4] PAR ERR H [2] — [2] WWPA 16 H [4]
BDOUT L — AE2 — [2] DOUT H [4,5]
BAD 17 L — AD1 — [2] A17 H [4]
[5] RPLY L [2/5] — [2] DIN L [2/5] — 8641 E1

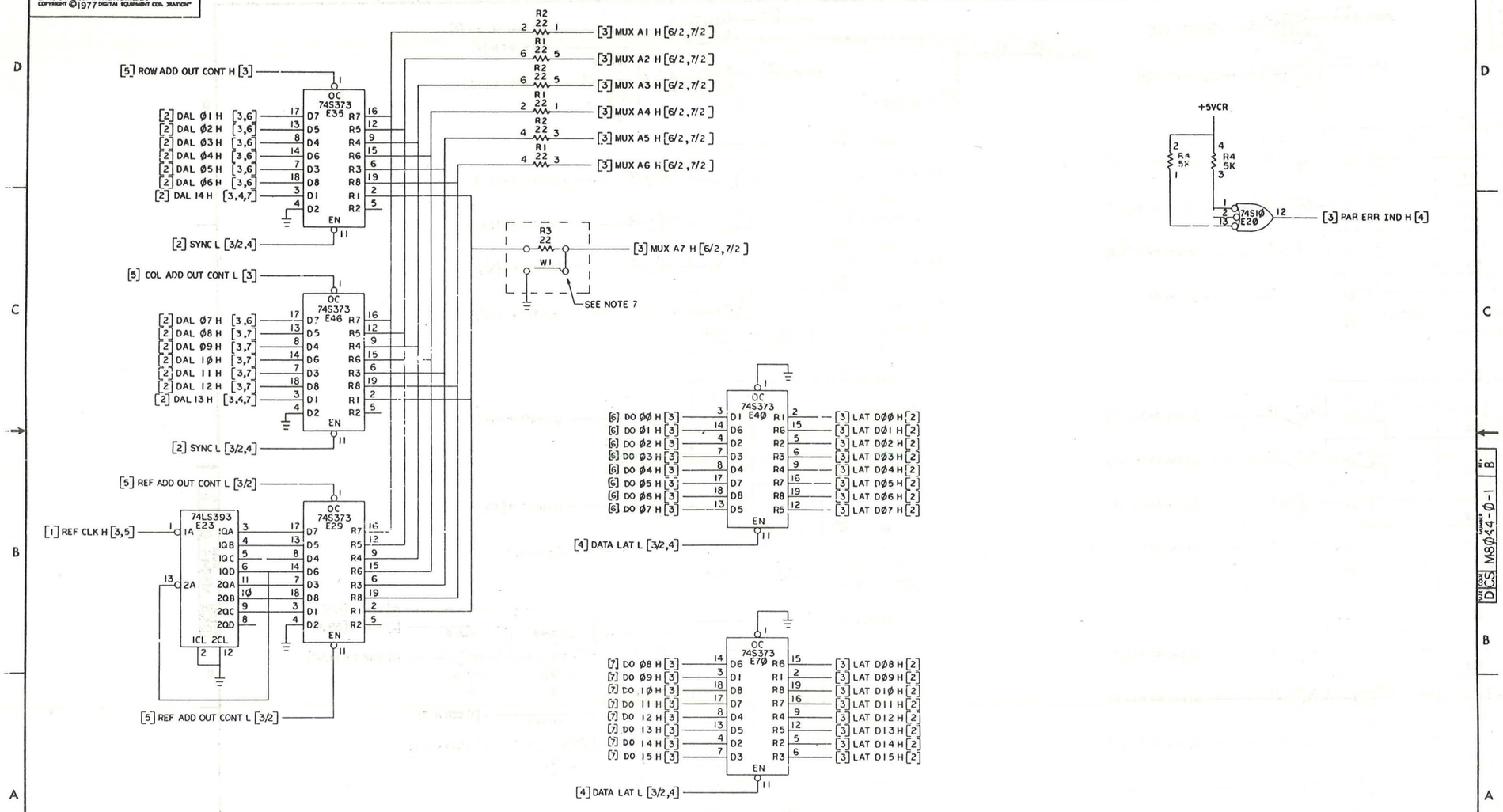
BSYNC L — AJ2 — [2] SYNC H [4/3]
[2] SYNC L [3/2,4]
74S240 E18 (1) — 74S240 E18 (19) — [2] SYNC 2 H [4,5/2]
[2] DIN H [5]
BDIN L — AH2 — 74S240 E18 (1) — [2] DIN L [2/5]
BRPLY L — AF2 — [5] RPLY H [2]
[5] RPLY EN L [2,5] — 8641 E8

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16BIT MOS MEM	D	CS	M8044-0-1	B
SCALE —	SHEET 2 OF 7	DIST.		

DEC FORM NO. DSD 138

"THIS DRAWING AND SPECIFICATION, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

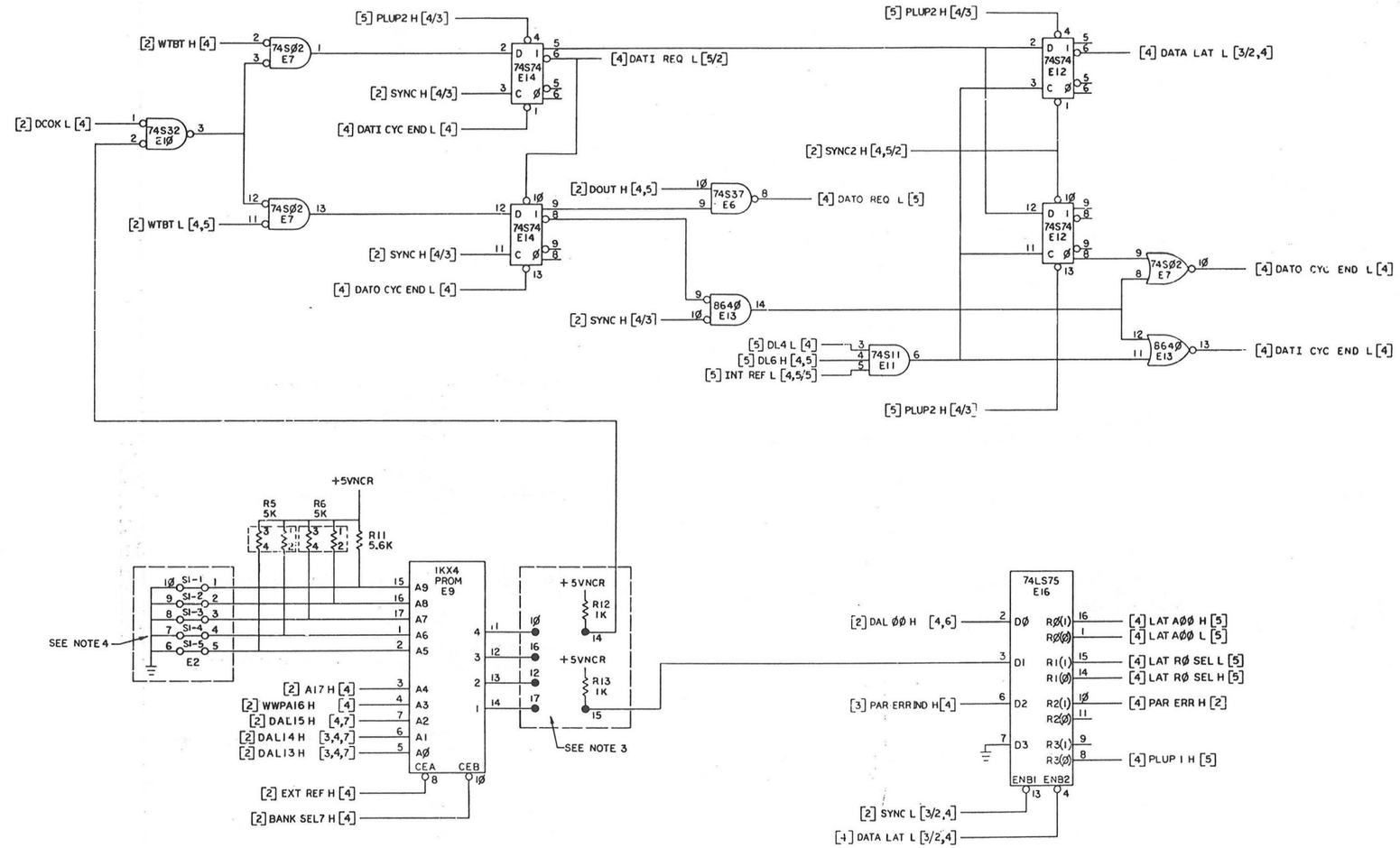


REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16BIT MOS MEM	D	CS	M8Ø44-Ø-1	B
SCALE — SHEET 3 OF 7	DIST.			

DEC FORM NO. 9 8 7 6 5 4 3 2 1

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

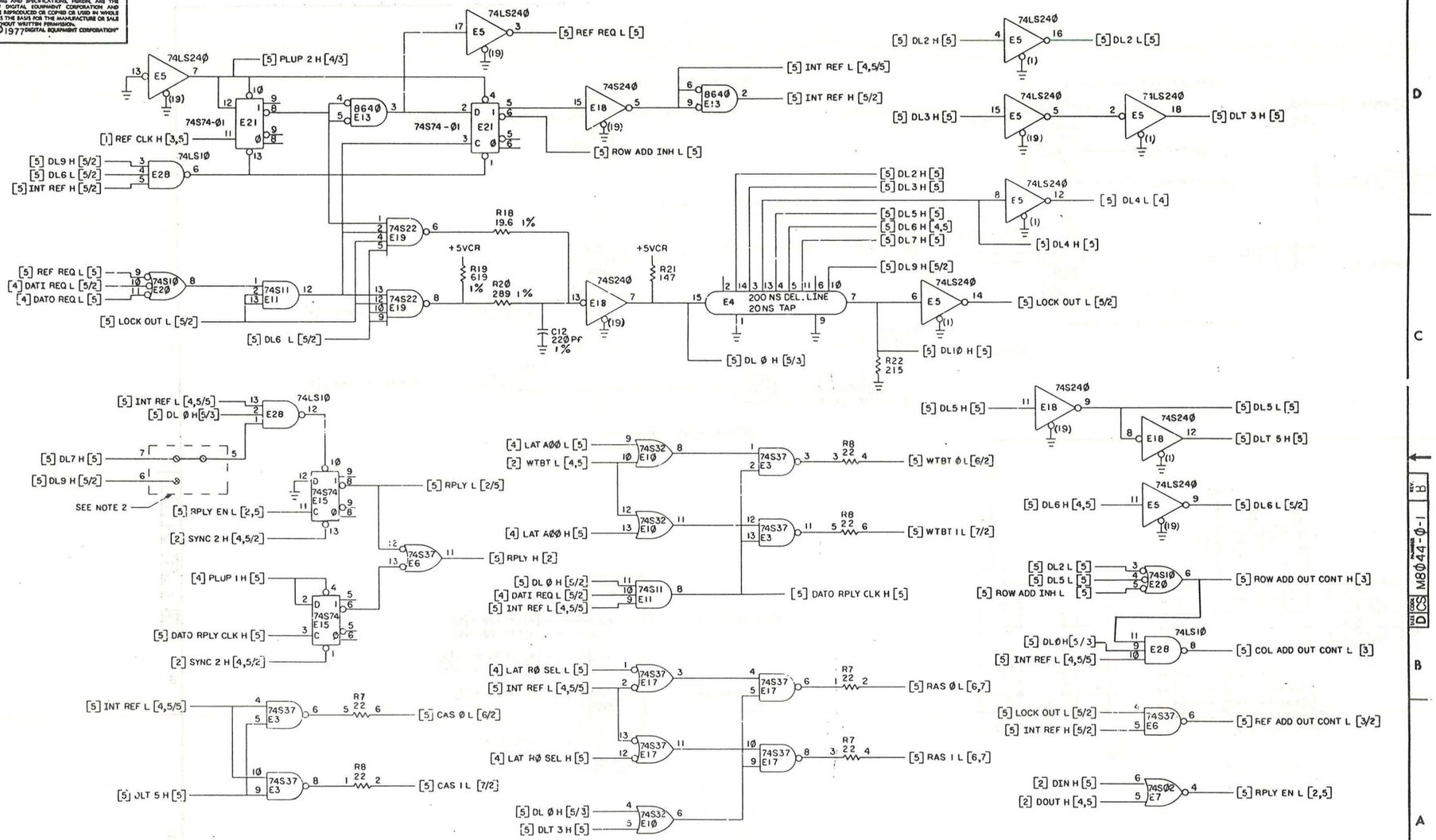


REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16 BIT MOS MEM	D	CS	M8Ø44-Ø-1	B
SCALE —	SHEET 4 OF 7	DIST.		

27

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

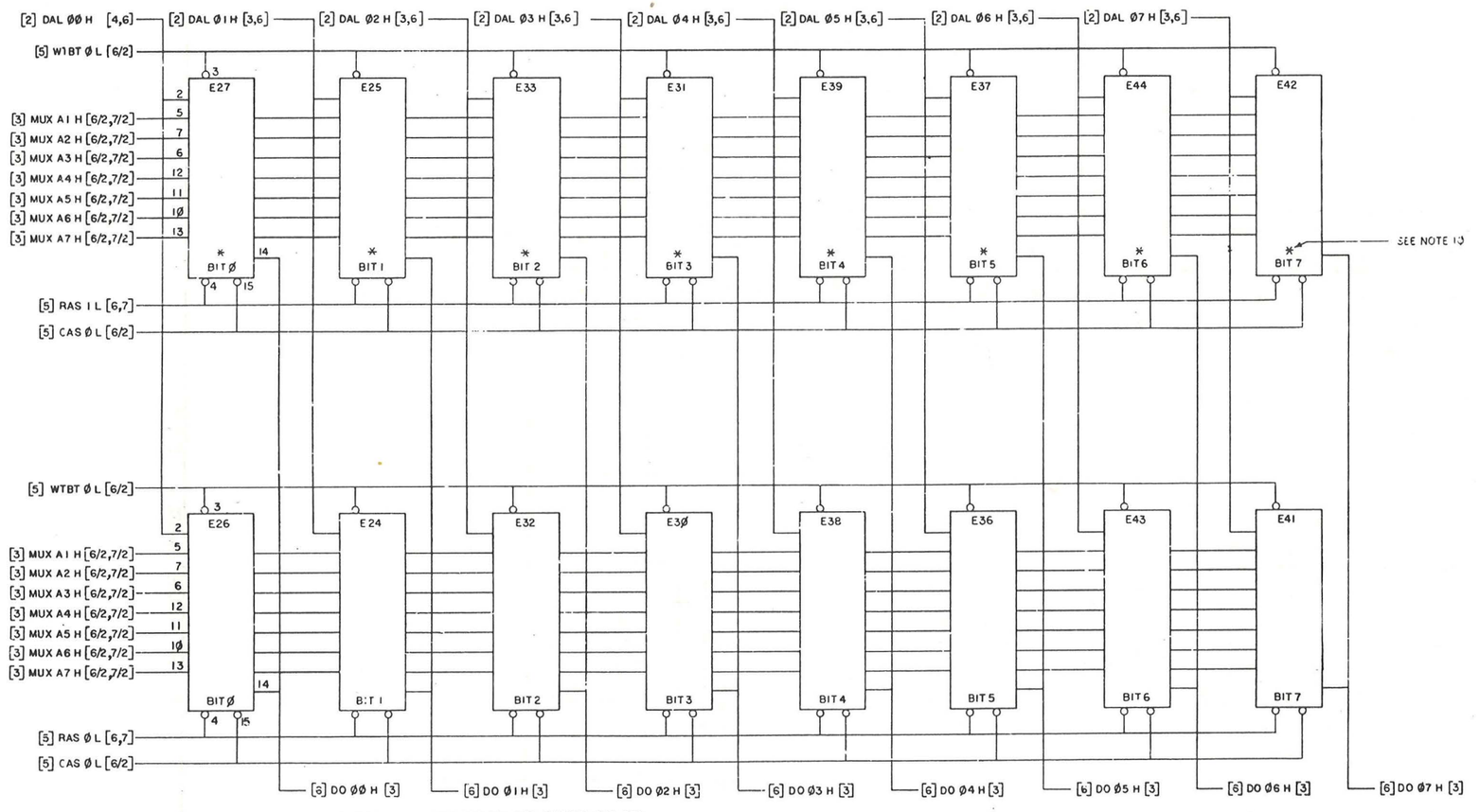


REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16 BIT MOS MEM	D	CS	M8Ø44-Ø-1	B
SCALE —	SHEET 5 OF 7	DIST.		

28

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

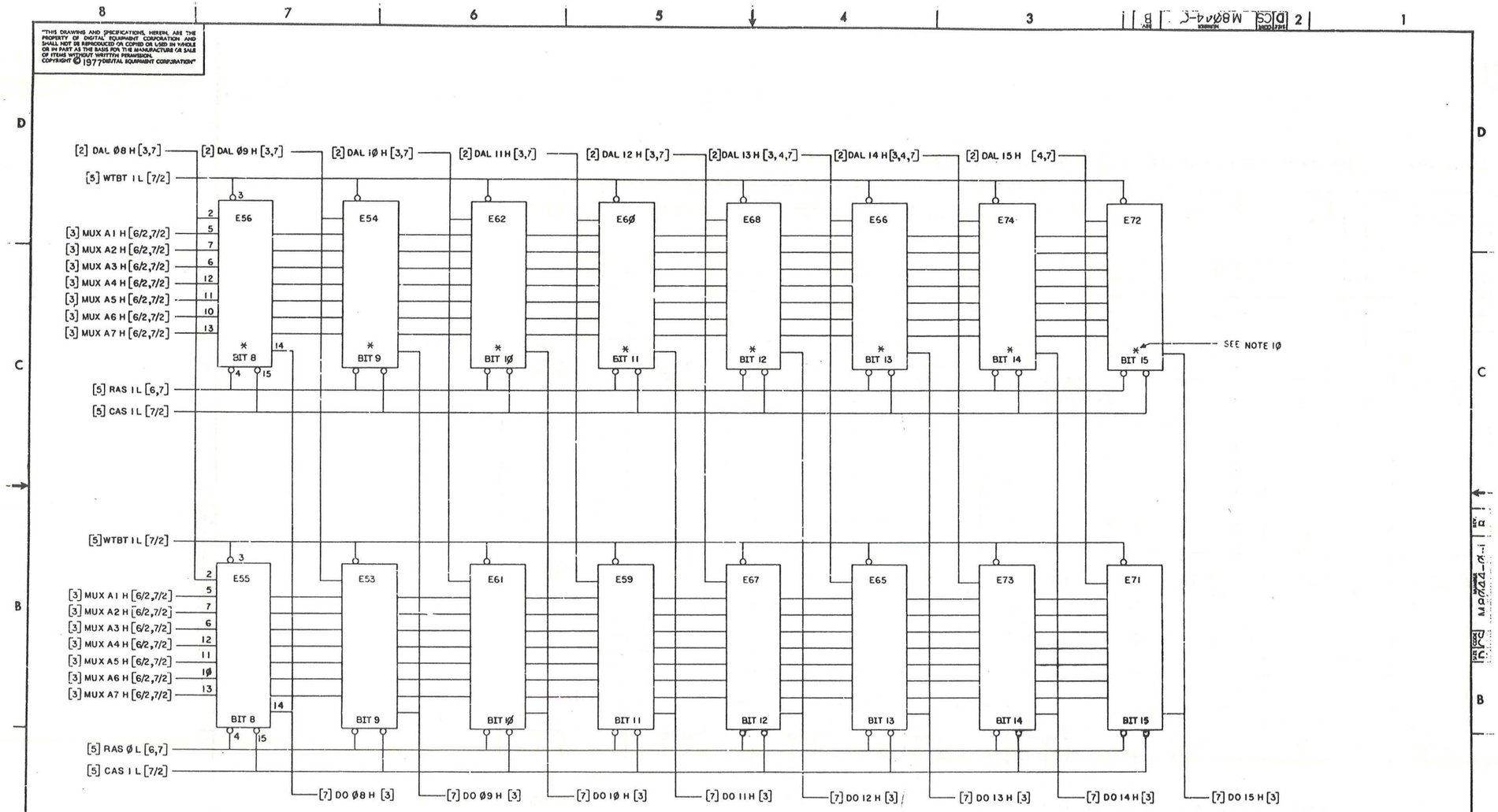


VCC1 — E26,E27,E32,E33,E38,E39,E43,E44, E55,E56,E61,E62,E67,E68,E73,E74
VCC2 — E24,E25,E3Ø,E31,E36,E37,E41,E42, E53,E54,E59,E6Ø,E65,E66,E71,E72
+5VNCR-E1, E8, E9, E15,E16,E45,E51,E52,E57,E58,E63,E64,E69,E7Ø
+5VCR—E3 THRU E7, E1Ø THRU E14, E17 THRU E23,E28,E29,E35,E4Ø,E46

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16BIT MOS MEM	D	CS	M8Ø44-Ø-1	B
SCALE —	SHEET 6 OF 7	DIST.		

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"



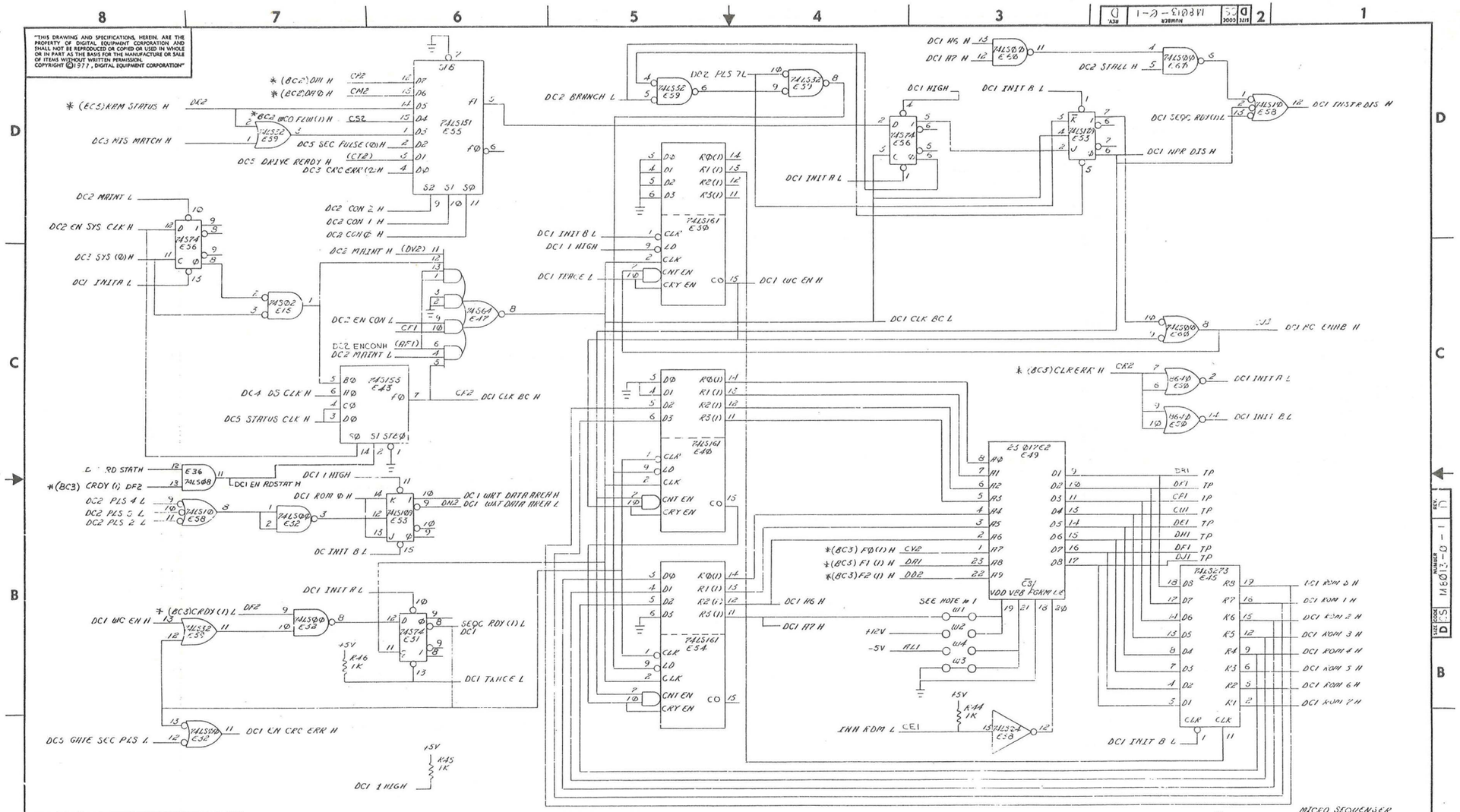
VCC1 — E26,E27,E32,E33,E38,E39,E43,E44, E55,E56,E61,E62,E67,E68,E73,E74
VCC2 — E24,E25,E3Ø,E31,E36,E37,E41,E42, E53,E54,E59,E6Ø,E65,E66,E71,E72
+5VNCR — E1, E8, E9, E15,E16,E45,E51,E52,E57,E58,E63,E64,E69,E7Ø
+5VCR — E3 THRU E7, E1Ø THRU E14,E17 THRU E23,E28,E29,E35,E4Ø,E46

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
32K 16 BIT MOS MEM	D	CS	M8Ø44-Ø-1	B
SCALE —	SHEET 7 OF 7		DIST.	

30

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977, DIGITAL EQUIPMENT CORPORATION"



NOTES:
1. FOR ROM PT. #23Ø17E2-ØØ INSERT W1 AND W3
   DELETE W2 AND W4.
   FOR ROM PT. #23Ø58B7-ØØ INSERT W2 AND W4
   DELETE W1 AND W3.
2. * NUMS REFERING TO POINT ON D CS- M8013-0-1

DRN.	
CHK'D	
ENG.	
PROJ. ENG.	
PROD.	
NEXT HIGHER ASSY.	D-UA-M8013-0-0
SCALE	NONE
SHEET	1 OF 5

FIRST USED ON: RLVII
TITLE: RLVII DISK CONTROL (DCI)
SIZE D | CODE CS | NUMBER M8013-0-1 | REV. D
DIST.

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

74LS175 E41
DC1 NPR DIS H
74LS08 E25
DU2
DC2 NPR EN H
DC2 RD DATA AREA H
DC2 EN LOOP LOCK H
DC2 EN LOOP LOCK L
DC2 EN DS CLK H
DC2 EN SYS CLK H
DC1 INIT B L
74LS175 E42
DC2 CON Ø H
DC2 CON 1 H
DC2 CON 2 H
DC2 BRANCH H
DC2 BRANCH L
74LS175 E26
DC2 TRACE HALT H
DV2
DC2 MAINT H
DC2 MAINT L
DC2 WRT GATE H
DC2 SEND DRY CMD H
74LS175 E27
CU2
DC2 DLY L
DC2 WRT CRC H
DC2 WRT MRK H
74LS175 E33
DC2 STALL H
DU2
DC2 DSK WRT H
DC2 WRT DAR H
DC2 RD STAT H
DC2 RD DATA AREA H
DU2
DC2 EN WRT RAM H
INITBL
74LS138 E46
DC1 ROM 5 H
DC1 ROM 6 H
DC1 ROM 7 H
DC2 PLS 5 L
DC2 PLS 4 L
DC2 PLS 2 L
DC1 INSTR DIS H
DC1 CLK BC L
DC1 WC EN H
74LS08 E36
DC3 CRC NOT ZERO L
74LS138 E37
DHC
DC2 SET CRDY L
DC2 SET OPT L
DC2 TST DCRC L
DB2
DC2 SYNC DA L
DC2 CLR MATCH L
DC2 TST MCRC L
DC1 I HIGH
MICRO SEQUENCER LATCHES
74LS08 E35
DU2
DC2 EN WRT RAM H
74LS32 E6
74LS32 E35
DC2 EN CON L
DC2 PLS 7 L
A5Y
A14
1K
EN CON H
BF1
* (BCSYRDY (1) L (DF2)
DC1 SEQC RDY (1) L
DC1 ROM 1 H
DC1 ROM 2 H
DC1 ROM 3 H
DC1 ROM 4 H

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
RLV11 DISK CONTROL (DC2)	D	CS	M8013-0-1	D
SCALE NONE SHEET 2 OF 5	DIST.			

DEC FORM NO. DRD 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

DC3 2 HIGH
74S74 E31
DC3 SYS (0) H
DISABLE CRYS L BD1
EXT CRYS L BC1
74S00 E20
74S00 E20
CD2 DC3 82 MHZ
+5V
X17 1K
X18 1K
Y1 8.2 MHZ
E12
74S37 E11
74S02 E15
74LS27 E21
74LS27 E21
E10 SOMS
X29 330 R/D
470
E18
74LS151 E9
DC3 WRT DATA PLS H
DC3 WRT DATA PLS L
WRT PRECOMP
25-2191A1 E8
74161 E12
DCI INIT B L
CLR
LD
CLK
CNT EN
CRY EN
CO
74LS86 E16
74LS174 E17
DCI INIT B L
(BC5) SER DMA OUT CN2
CRC GENERATOR
9401 E3
DC3 CRC (1) H
DC3 CRC NOT ZERO H
DC3 CRC NOT ZERO L
DC3 MUX DATA L
74S64 E22
74S04 E6
8640 E50
74S04 E6
DC3 WC EN H
DC2 WRT MKR H
DC2 WRT DIR H
DC2 MAINT L
DC4 DS DATA (1) H
DC2 EN LOOP LOCK H
(BC2) DH (1) L CB2
74LS04 E38
74LS27 E21
74LS08 E36
74LS02 E18
DCI WRT DATA AREA H
DC4 MARKER (1) L
DCI CLK BC L
74LS04 E38
74LS32 E35
74LS08 E36
DC2 SET OPI L
DMR
DC3 PRESET OPI L
DC3 CRC ERR (1) H
DC3 CRC ERR (3) H
74LS74 E4
74LS32 E32
DC2 TST MCRC L
DC2 TST DCRC L
DCI INIT B L
DC3 2 HIGH
DC2 BRANCH H
DC2 EN DS CLK H
DCI DS CLK H
DCI INIT B L
74LS08 E25
74LS112 E7
74LS112
74LS86 E16
74LS42 E15
DCI WRT DATA AREA H
DC2 CLR MATCH L
DC3 MISMATCH L
DC3 MIS MATCH H
HEADER MATCH DETECTION

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
RLV11 DISK CONTROL (DC3)	D	DCS	MS013-0-1	D
SCALE NONE SHEET 3 OF 5	DIST.			

DEC FORM NO. DRD 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

READ DATA PULSE GENERATOR

DC2 MAINT H DV2
DC3 WRT DATA PLS L
74S37 E11
74S37 E11
E2 100NS
R23 220
74S12 E14
DC4 3 HIGH
DC5 READ DATA H
74LS112 E28
+5V
R51 1K
DC4 3 HIGH
DC2 EN LOOP LOCK L
74S04 E6
74S04 E24
74S10 E29
74S20 E29
74S10 E29
74S20 E29
74S10 E14
DC4 EARLY ERROR L
PHASE/FREQUENCY ERROR DETECTION
74S04 E24
74S10 E14
DC4 LATE ERROR L
74S04 E24

READ DATA SEPARATOR

DC4 3 HIGH
DC2 EN DS CLK H
74S74 E23
DC1 INIT B L
74S74 E13
DC4 3 HIGH
74S74 E25
DC1 DS ENABLE H
DC4 3 HIGH
74LS393 E49
DC4 MARKER H
DC4 MARKER L
* (@C5) CRDY (1) L
74S00 E5
74S02 E18
DC4 DS CLK H
74S00 E5
74S02 E18
DC4 MARKER L
74S00 E5
EXT PLL CLK L
INH VCO CLK L
+5V
R32 1K
R33 1K
TP2 VCO
TP3 GND

+12V
D15 1N 6.8 BZI 6.8V
R31 464 1%
R42 1K
Q5 2N4403
D6 1N4154
D7 1N4154
D9 1N4154
D10 1N5.8BZI 6.8V
R34 1.62K 1%
R50 1K
Q6 2N4403
TP1 VCI
D11 1N5.1BZI
K21 180 1/2W
K22 180 1/2W
K23 180 1/2W
K24 180 1/2W
D12 1N5.1BZI
DC4 LATE ERROR L
Q7 2N4403
DC4 EARLY ERROR L
Q8 2N4403
R33 348 1%
R35 1.1K 1%
C62 47PF
74S124 E19
Q9 2N4401
R20 130
C63 100PF
D9 1N4154
R17 100
R15 120 1%
C61 .01MF
R32 1.33K 1%
R41 5K
C68 .22MF
R54 10K 1/2W
-10V
D5 6.8V 1N6.8BZI
C45 .22UF
-5V
VCO CONTROL - CHARGE PUMP AND FILTER

+12V
R43 5.1K
C66 5600PF
C60 .01MF
R37 2.2K
CTL VTH DCG
555 E1
R19 120
D13 1N754A
Q3 MPS93
C72 .1MF
C73 .1MF
D14 1N754A
Q1 MPS05
C67 .047UF
R27 220
D1 1N752
D2 1N752
K1 2-7
C74 15 UF
C75 .1 UF
-10V AND -5V CHARGE PUMP

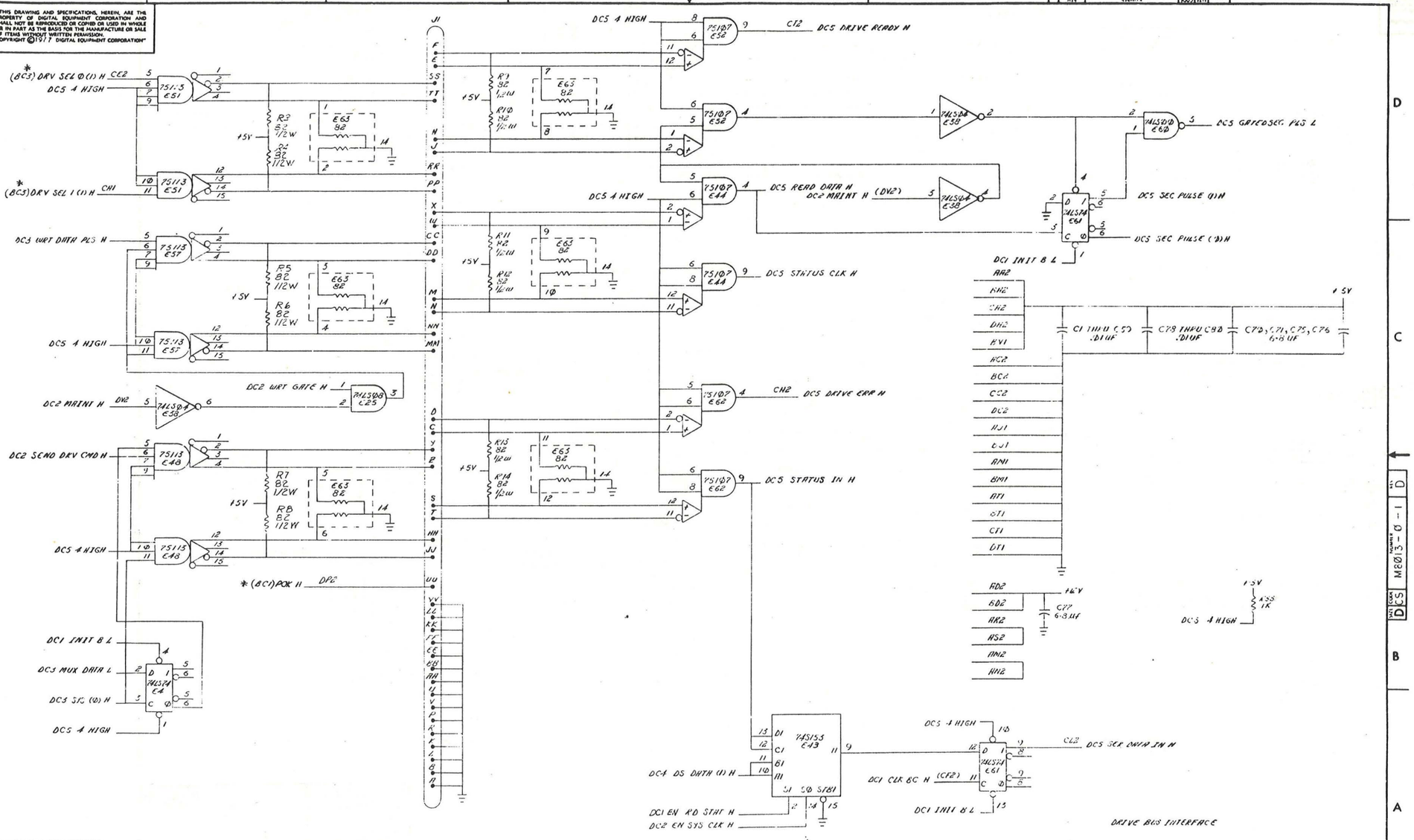
REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
RLV11 DISK CONTROL DC4	D	CS	M8013-C-1	D
SCALE NONE	SHEET 4 OF 5	DIST.		

DEC FORM NO. 040 138

34

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

TITLE		SIZE	CODE	NUMBER	REV.
RLV11 DISK CONTROL	DC5	D	S	MECS-X-1	D
SCALE NONE	SHEET 5 OF 15	DIST			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977, DIGITAL EQUIPMENT CORPORATION"

BC5 DIS BUS XFR (I) L
BC5 DIS BUS XFR (I) L
BC5 LIMIT (I) H
BC5 LIMIT (I) H
BC5 (B) HIGH
BC5 (A) HIGH
BC5 (D) HIGH
BC5 (C) HIGH
BC2 MATCH H
BC2 TS DAL 2 H
BC2 TS DAL 1 H
BC2 SC RDY (I) L
BC3 VEC TO BUS H
BC3 INIT L
BC3 INIT H
BC3 1ERDY H
BC3 CLR ERR L
BC3 5.2 MHZ H
BC2 NPR EN H
BC2 COMP ERR L
BC2 WC OFLW (I) H
BC2 NPR EN H
BC2 DSK WRT H
BC3 INIT H
BC3 NXM (I) H
BCI GS RPLY H
BCI M RPLY H
BCI R SYNC L
BCI R SYNC H
BCI R DOUT L
BCI R DOUT H
BCI R DIN H
BCI R DIN L
BCI B POK L
BCI MASTER L
BCI MASTER RPLY L
BCI DSK WRT L
BCI G DIN H
BCI G DOUT H
BCI G DOUT PLS L
BCI G ADD H
BCI G EN ADD H
BCI EN ADD H
BCI EN DATA L
BCI M DIN H
BCI SYNCO H
BCI DAL REC H
BCI DAL XMIT EN H
BCI SEL CSR L
BCI SEL BA L
BCI SEL DA L
BCI GD OUT L
BCI SEL MP L
BCI WRT CSR L
BCI WRT CSR PLS H
BCI RD CSR L
BCI RD BA L
BCI WRT BA L
BCI WRT DA L
BCI WRT CSR PLS L
BCI WRT DA PLS L
BCI RD DA L
BCI RD FIFO L
BCI GS RPLY H
BCI NXM (I) H
BCI NXM (I) L
B SRPLY L
BNI B SACK L
AK2 BWTBT L
AN1 B DMR L
A32 B DMGO L
AF2 B RPLY L
AJ2 F SYNC L
AE2 F DOUT L
AH2 EDIN L
AL2 BIRQ L
BBI B P OK H
B P CK H
EFI
CPI
BCI P CK H
DU1
CD1
DJ1
DMA CONTROL
DC0I0 E7
BDMGI
REQ
DATIN
DATIO
CNTEN
BUSYNC
RPLY
CLK
INIT
BDMR
BDMGO
MASTER
SYNC
DIN
DOUT
ADREN
DATEN
TMOUT
DC0X4 E12
ENB
RXCX
SYNC
DAL 2
DAL 1
DAL 0
WTBT
DOUT
DIN
VECTOR
BRPLY
SEL 6
SEL 4
SEL 2
SEL 0
CUT HB
CUT LB
INWD
BUS PROTOCOL
DEV SEL
WAITING
NXM DETECTION
10µS MIN
R1 1K
R2 1.1K
R3 330
R4 680
R5 3.48K 1%
R6 330
R7 680
R8 100
R9 1K
R10 1K
R11 4.64 1%
R12 51K
C1 4700 PF 1%
C2 150 PF
C3 220 PF
C4 1000 PF
Q1 MXA05
Q2 MXA55
E641 E1
E641 E14
74LS04 E9
74LS08 E2
74LS00 E50
74LS11 E74
74S74 E15
74LS24 E9
74S04 E34
74LS00 E43
74LS02 E4
74LS27 E57
74LS11 E11
74LS02 E24
74LS32 E3
74LS20 E24
74LS74 E10
74LS02 E52
74LS04 E49
74LS00 E24
8881 E8
74LS10 E44
74LS28 E2
74LS22 E4
7432 E19
74LS32 E29
7432 E13
7432 E3
74LS22 E3
74LS04 E9
74LS04 E34
74LS74 E6
74LS22 E5
74LS123 E20
74LS74 E6
74LS04 E34
74LS08 E2
74LS02 E4

NOTE:
1. * MEANS REFERRING TO POINT ON D-CS-MBC13-0-1.

REVISIONS			
CHK	CHANGE NO.	REV.	
	74-[illegible]-CN-C001	B	
	[illegible]		R. HULTMAN
	[illegible]		
	M8014-0000-5	C	
	M8014-0000-5	C	R. HULTMAN

DRN.		FIRST USED ON: RLV11
CHK'D		digital
ENG.		TITLE: RLV11 BUS CONTROL (SCD)
PROJ. ENG.		
PROD.		
NEXT HIGHER ASSY.		SIZE: D CODE: CS NUMBER: M8014-0-1 REV: C
SCALE		SHEET 1 OF 5 DIST.

DEC FORM NO. 5000 1020-0

36

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

BCI SEL BA L
BC3 INIT L
74LS11 E11
BCI SEL MP L
BUS XCVRS
BDAL 13 L BT2
BDAL 12 L BS2
BDAL 11 L BR2
BDAL 14 L BU2
DC005 E33
+5V
BBS 7 L AP2
E23 ON=1
AI2 AI1 AI0 A9 A8 A7 A6 A5 A4 A3
WC/BA LO BYTE
DC006 E18
WC/BA HI BYTE
DC006 E25
CNT IA CLK-A S-A CLK-C S-C RD LD RD-A
128 D/F 64 D/F 32 D/F 16 D/F 8 D/F 4 D/F 2 D/F 1 D/F MAX-A MAX-C
BCI GEN ADD H
BCI RD BA L
BCI G DOUT FLS L
BCI EN ADD H
BC5(I) HIGH
74LS74 E35
BC3 CLR ERR L
BCI RD CSR L
8556 E17 XBA
BCI WRT CSR L
BC2 INIT A H
74LS02 E51
BCI WRT CSR PLS H
BCI EN ADD H
BDAL 10 L BP2
BDAL 9 L BN2
BDAL 8 L BM2
BDAL 15 L BV2
DC005 E27
V8
BDAL 7 L BL2
BDAL 6 L BK2
BDAL 5 L BJ2
BDAL 0 L AU2
DC005 E21
R19 261 1%
+5V
E22 ON=1
V7 V6 V5 V4 V3 V2
BC3 VEC TO BUS H
BDAL 4 L BH2
BDAL 3 L BF2
BDAL 2 L BE2
BDAL 1 L AV2
DC005 E13
BC2 MATCH H
BCI DAL XMIT EN H
BCI DAL REC H
BC2 TS DAL 13 H
BC2 TS DAL 12 H
BC2 TS DAL 11 H
BC2 TS DAL 14 H
BC2 TS DAL 10 H
BC2 TS DAL 9 H
BC2 TS DAL 15 H
BC2 TS DAL 7 H
BC2 TS DAL 6 H
BC2 TS DAL 5 H
BC2 TS DAL 0 H
BC2 TS DAL 4 H
BC2 TS DAL 3 H
BC2 TS DAL 2 H
BC2 TS DAL 1 H
BC2 WC OFLW (1) H
BC2 WC OFLW (0) H
8556 E17 XBA
B DAL 17X
B DAL 16X
BCI RD DA L
BCI WRT DA L
8556 E47
8556 E48
8556 E36
8556 E28
DISK ADDRESS REGISTER
BCS(F) HIGH
BCS(I) HIGH
DISK ADDRESS SERIALIZER
STB
7415Ø E45
10 CBI BC2 DA (1) L
BC5 BCNTA 3 H
BC5 BCNTA 2 H
BC5 BCNTA 1 H
BC5 BCNTA 0 H
* (DC2) INC DA L
BCI WRT DA PLS L
7408 E56
BC3 INIT L
74S04 E34
CPI BC2 DA 1 H
CMI BC2 DA 0 H
BCI INIT A H

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
RLV11 BUS CONTROL (BC2)	D	CS	M8014-0-1	C
SCALE —	SHEET 2 OF 5	DIST.		

DEC FORM NO. DRD 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

INTERUPT CONTROL
DC003
E16

BC3 IE RDY H
BIAKI L AM2
BINIT L AT2
BCI R DIN L
RQSTB
ENA DAT
ENE DAT
ENA CLK
ENA ST
BIRQ
BIAKO
INITO
VECTOR
VEC RQST
ENB ST
ENB CLK
AN2 BIAKO L
BC3 INIT L
BC3 VEC TO BUS H

R13 330Ω +5V
R14 680Ω
R15 330Ω +5V
R16 680Ω
R23 1K +5V

BC5(B)HIGH
BC3 C RDY(I) H
BC3 INIT H
74LS02 E5
74LS74 E10
BC3 IERDY H

BCI WRT CSR PLS H
74LS174 E26
R20 10K R21 10K R22 10K +5V
BC2 TS DAL 1 H
BC2 TS DAL 2 H
BC2 TS DAL 3 H
BC2 TS DAL 6 H
BC2 TS DAL 8 H
BC2 TS DAL 9 H
D3 D2 D1 D0 D4 D5 CLK
R3(I) R2(I) R1(I) R0(I) R4(I) R5(I) CLR
CVI BC3 FO (I) H
DCI BC3 F1 (I) H
DDI BC3 F2 (I) H
CEI BC3 DRV SEL Ø (I) H
CCI BC3 DRV SEL 1 (I) H

DC2 SET C RDY L DHI 1Ø 74LS08 E52
BC5(G) HIGH
BC3 COMP ERR H
74LS74 E63
DFI BC3 C RDY (I) L
BC3 C RDY (I) H
BCI WRT CSR PLS L
BCI TS DAL 7 H
74LS32 E29
74LS04 E56
74LS04 E62
12 CRI
BC3 CLR ERR H
BC3 CLR ERR L

BCI RD CSR L
8097 E70 (I)

BC3 C RDY(I)H CH1
DC5 DRV ERR H
DC3 CRC ERR(I) H DL1
BCI NXM(I) L
74LS51 E40
74LS10 E44
74LS04 E49
BC3 COMP ERR L
BC3 COMP ERR H
BC2 TS DAL 15 H
DC5 DRIVE READY H DT2
BC2 TS DAL Ø H

+5V R17 180K C5 6.8UF
DC2 INC DA L DBI 2
74LS123 E20
DC2 DLT L CUI 4
74LS08 E52
DC3 PRESET OPI L DMI
74LS109 E69
BC5(G) HIGH
BC3 CLR ERR L

BC3 SCRDY(I) L
BCI R SYNC L
BC5(C) HIGH
74LS74 E15

DC3 MIS MATCH L CKI
74LS02 E51

BC3 INIT H
BC2 TS DAL 7 H
BC2 TS DAL 1Ø H
BC2 TS DAL 12 H
BCI NXM (I) H
BC2 TS DAL 13 H
BC2 TS DAL 11 H
BC2 TS DAL 14 H

8097 E76 (I) (15)
8097 E30 (I) (15)

REVISIONS		
CHK	CHANGE NO.	REV

TITLE	SIZE	CODE	NUMBER	REV.
RLV11 BUS CONTROL (BC3)	D	CS	M3014-Ø-1	C
SCALE —	SHEET 3 OF 5	DIST.		

DEC FORM NO. 1-90

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT ©1977 DIGITAL EQUIPMENT CORPORATION"

BC1 RD FIFO L
BC1 EN DATA L
74LS00 E50

BC5 LD BOUT PLS L

FIFO OUTPUT BUFFER

BC5 ED BIN H
BC1 MDIN H

74LS295 E65: BC2 TS DAL 6 H, BC2 TS DAL 9 H, BC2 TS DAL 10 H, BC2 TS DAL 11 H → BC2 TS DAL 12 H, BC2 TS DAL 13 H, BC2 TS DAL 14 H, BC2 TS DAL 15 H
74LS295 E31: BC2 TS DAL 4 H, BC2 TS DAL 5 H, BC2 TS DAL 6 H, BC2 TS DAL 7 H → BC2 TS DAL 8 H, BC2 TS DAL 9 H, BC2 TS DAL 10 H, BC2 TS DAL 11 H
74LS295 E77: BC2 TS DAL 0 H, BC2 TS DAL 1 H, BC2 TS DAL 2 H, BC2 TS DAL 3 H → BC2 TS DAL 4 H, BC2 TS DAL 5 H, BC2 TS DAL 6 H, BC2 TS DAL 7 H
74LS295 E38: → BC2 TS DAL 12 H, BC2 TS DAL 13 H, BC2 TS DAL 14 H, BC2 TS DAL 15 H
74LS295 E64: → BC2 TS DAL 8 H, BC2 TS DAL 9 H, BC2 TS DAL 10 H, BC2 TS DAL 11 H
74LS295 E54: → BC2 TS DAL 4 H, BC2 TS DAL 5 H, BC2 TS DAL 6 H, BC2 TS DAL 7 H
74LS295 E71: → BC2 TS DAL 0 H, BC2 TS DAL 1 H, BC2 TS DAL 2 H, BC2 TS DAL 3 H

BC5(L) HIGH
FIFO INPUT BUFFER
BC5(E) HIGH
BC5(L) HIGH
BC5(F) HIGH
BC5(G) HIGH
BC5(M) HIGH
BC5(C) HIGH

BC4 TS FIFO 0 H
BC4 TS FIFO 1 H
BC4 TS FIFO 2 H
BC4 TS FIFO 3 H
BC4 TS FIFO 4 H
BC4 TS FIFO 5 H
BC4 TS FIFO 6 H
BC4 TS FIFO 7 H
BC4 TS FIFO 8 H
BC4 TS FIFO 9 H
BC4 TS FIFO 10 H
BC4 TS FIFO 11 H
BC4 TS FIFO 12 H
BC4 TS FIFO 13 H
BC4 TS FIFO 14 H
BC4 TS FIFO 15 H

BC5 ED BIN H
BC1 M DIN H

74LS295 E32: BC2 TS DAL 12 H, BC2 TS DAL 13 H, BC2 TS DAL 14 H, BC2 TS DAL 15 H

BC5(E) HIGH

(RD/WRT ADDR CNTR)
DC0Ø6 E60
74LS04 E49 ← BC3 CLR ERR L
BC5 INC RA H
BC5(K) HIGH
BC3 EO RAM L
BC5 WRT RAM L

2111A2 E78, E66, E73, E72

BC5(G) HIGH
DC5 SER DATA IN H
74LS74 E68 (SAVE)
BC3 INIT L

WRT/RD SHIFT REGISTER
74LS299 E55 ODD
74LS299 E61 EVEN
BC5(K) HIGH
BC5(J) HIGH
BC5 SR LD EN H
74LS04 E34 ← BC4 BCNTA Ø L
BC3 INIT L
BC5 WRT RAM L (DJ1)
DC2 DSK WRT H
BC5 BCNTA Ø H

BC4 BCNTA Ø L
74LS51 E67
74LS27 E57
BC4 SER DATA OUT H
BC5 ...PTY Ø H
DC WRT DATA AREA L
DNI

(FIFO BUFFER)

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE		SIZE	CODE	NUMBER	REV.
RLV11 BUS CONTROL (BC4)		D	CS	M8014-0-1	C
SCALE	SHEET 4 OF 5	DIST.			

DEC FORM NO. DRD 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

AC2 BC2 CC2 DC2 AJ1 BJ1 AM1 BM1 AT1 BT1 CT1 DT1

AD2 BD2 +12V

AA2 BA2 CA2 DA2 BV1 +5V

BC5(G) HIGH

DC2 EN WRT RAM H

74LS74 E68

74LS51 E67

BC5 WRT RAM L

BC3 CLR ERR L

BCI DSK WRT L

74LS74 E35

74LS32 E29

BC5(I) HIGH

74LS175 E43

BC5 BCNTA 2 H
BC5 BCNTA 1 H
BC5 BCNTA 0 H
BC5 BCNTA 3 H

BC5 SR LD EN H

BC5 SYNK H

BC5 SYNK L

74LS10 E44

74LS11 E74

74LS175 E42

BCI WRT DATA AERA L (DH1)

BC5 EO RAM H

BC5 EO RAM L

BC5 WRT INC RA H

BC5 EO BIN H

BC5 MAX MIN L

BC5 LD BOUT PLS L

74LS04 E62

74LS00 E39

74LS20 E39

BC5 WRT EMPTY (0) L

74LS08 E56

BC5 EN RAM XF (I) H

74LS161 E46

CLR LD CNT EN CRYEN CO CE

DC1 CLK BC H CF1

74LS04 E34

DC2 INC DA L (DB1)

DC1 EC ENAB H (DJ1)

BC5(H) HIGH

DSK WRT H (DJ1)

DC2

BC3 CLR ERR L

BC5 EN ROM L

R18 1K

+5V

BC5 LIMIT (I) L

DC2 DSK WRT H (DJ1)

DC2 MAINT H (DV1)

BC3 C RDY (I) H

74LS27 E57

74LS00 E50

BC2 WC CFLW (I) H (DD1)

BCI MASTER CLR PLY L

BCI RD FIFO L

74LS03 E52

74S74 E59

BC5 DIS BUS XF (I) H

BC5 DIS BUS XF (I) L

BC5 SYNK H

74S08 E56

BC5 MAX MIN L

BC5(F) HIGH

BC5 LIMIT (I) H

BC5 LIMIT (I) L

74LS02 E75

74LS32 E75

S DR1 BC5 RAM STATUS H

BC5 SYNK L

BC3 CLR ERR L

DC2 MAINT H (DV1)

74LS74 E37

74LS24 E62

BC5(F) HIGH

BC5 WRT EMPTY (0) L

BC5 WPT EMPTY (0) H

BC3 INIT L

BCI DSK WRT L

BC2 WCO FLW(0) L

74LS22 E51

74LS51 E40

DC2 DSKWRT H (DJ1)

BC5 INC RA H

74191 E59

74191 E53

MAX/MIN RCLK D3 D2 D1 D0 R3(I) R2(I) R1(I) R0(I) LD DN/UP CLK ENB

BC5 EN ROM L

FIFO WORD COUNTER

BC3 CLR ERR L

BC5 EO RAM H

+5V

C10 – C35 .047UF

C6 – C9 6.8UF

R24 1K R25 1K R26 1K R27 1K R28 1K R29 1K R30 1K R31 1K R32 1K R33 1K R34 1K R35 1K R36 1K

BC5(A) HIGH BC5(B) HIGH BC5(C) HIGH BC5(D) HIGH BC5(E) HIGH BC5(F) HIGH BC5(G) HIGH BC5(H) HIGH BC5(J) HIGH BC5(K) HIGH BC5(L) HIGH BC5(M) HIGH

REVISIONS

CHK	CHANGE NO.	REV.

TITLE: RLV 11 BUS CONTROL (BC5) (FIFO COUNTROL)

SCALE: SHEET 5 OF 5

SIZE	CODE	NUMBER	REV.
D	CS	M8014-0-1	C

DIST.

DEC FORM NO. DRD 138

40

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 , DIGITAL EQUIPMENT CORPORATION"

74LS174 E43
DI TS BUS 9 H — D5 — R5(I)
DI TS BUS 8 H — D4 — R4(I)
DI TS BUS 4 H — D3 — R3(I)
DI TS BUS 3 H — D2 — R2(I) — DI FUNCT 2 H
DI TS BUS 2 H — D1 — R1(I) — DI FUNCT 1 H
DI TS BUS 1 H — DØ — RØ(I) — DI FUNCT Ø H
CLR CLK
D2 INIT L
D3 WRT CSR PLS H

D4 BTR H — 8Ø97 E39 — DI TS BUS 7 H
DI TS BUS 9 H — 8Ø97 E39 — DI TS BUS 9 H
DI TS BUS 8 H — 8Ø97 E39 — DI TS BUS 8 H
D4 B ERROR H — 8Ø97 E39 — DI TS BUS 15 H
8Ø97 E38 — DI TS BUS 12 H
DI TS BUS 4 H — 8Ø97 E38 — DI TS BUS 4 H
8Ø97 E38 — DI TS BUS Ø H
8Ø97 E38 — DI TS BUS 13 H
D3 IE H — 8Ø97 E39 — DI TS BUS 6 H
D3 RD CSR L
D4 B DONE H — 8Ø97 E39 — DI TS BUS 5 H
+3V2 — 8Ø97 E38 — DI TS BUS 11 H
8Ø97 E38 — DI TS BUS 14 H

74LS299 E37
D2 INIT L — CLR
D3 DB CLK H — CLK
D4 SER DATA H — SR
SL
D2 LO SEL Ø H — SØ
D2 DB SEL 1 H — SI
ENA 1
ENA 2
DI TS BUS 7 H
DI TS BUS 5 H
DI TS BUS 1 H
DI TS BUS Ø H
DI TS BUS 2 H
DI TS BUS 4 H
DI TS BUS 6 H
DI DATA 7 H

DI TS BUS 3 H — 8Ø97 E42
D2 DB SEL 1 L

74LS257 E34
+3V2 — B3, A3, B2, A2
D4 RX AC LO H — B1
B1
D3 NXM (1) H — BØ
AØ
OUT CONT SØ
F3, F2, F1 — DI TS BUS 3 H
FØ — DI TS BUS 1 H
DI EN MUX L
D4 B DONE H

74LS299 E33
CLR
CLK
D2 SHFT RT H — SR
SL
D2 HI SEL Ø H — SØ
SI
ENA 1
ENA 2
DI TS BUS 15 H
DI TS BUS 13 H
DI TS BUS 9 H
DI TS BUS 8 H
DI TS BUS 1Ø H
DI TS BUS 12 H
DI TS BUS 14 H
DI DATA 15 H

74LSØ8
D4 ENA 12 BIT L — E22
D3 RD CB L — E22
D3 EN DATA L — E22
DI EN MUX L

DI TS BUS 11 H — 8Ø97 E42 — DI DATA 11 H
D2 DB SEL 1 L

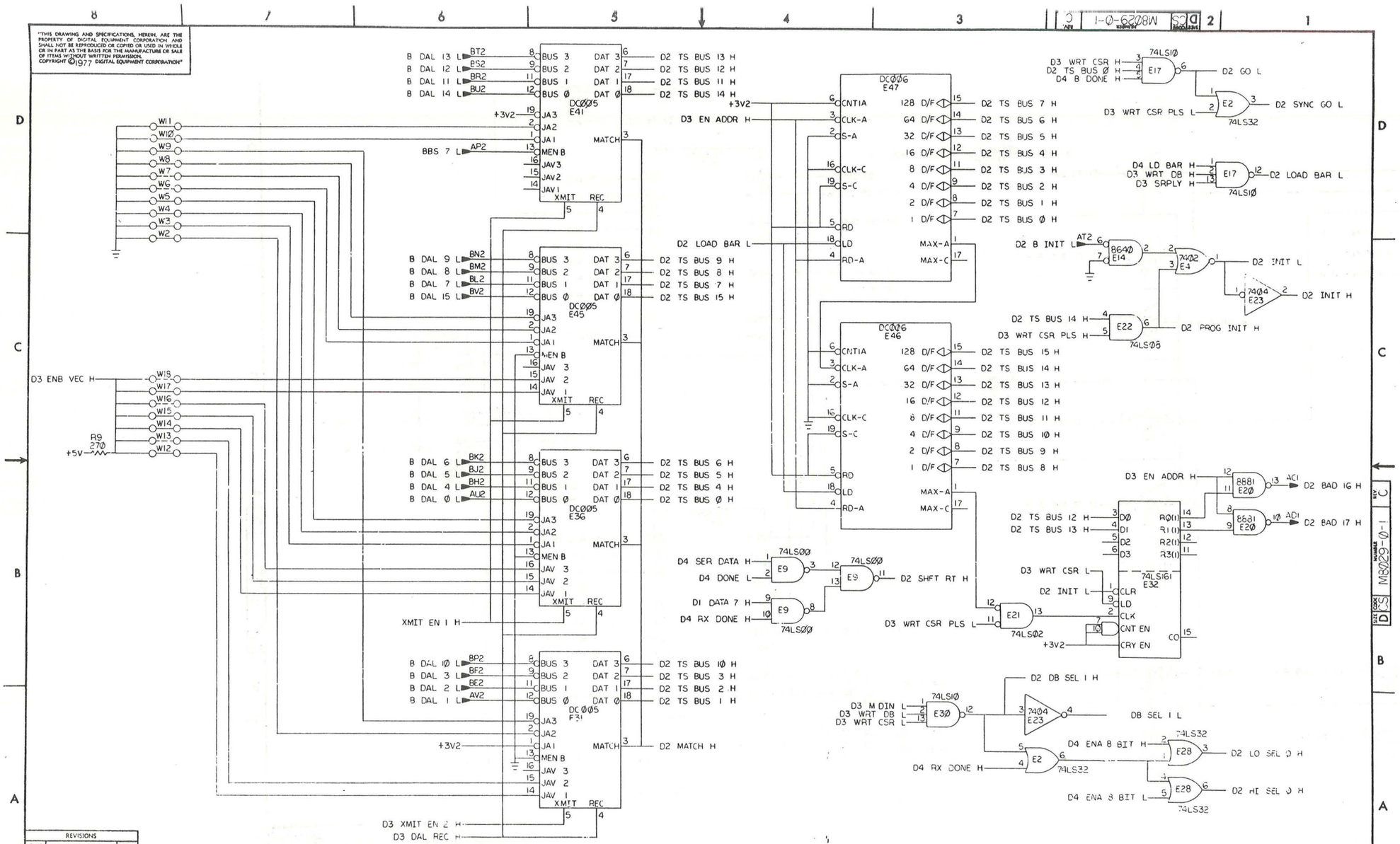
REVISIONS			
DIR	CHANGE NO.	REV.	
1M1	M6129-C0001	C	
	J.GLAVIN		

DRN: Larry M. Cole 10/25/77
CHK'D Robert Thomas 12/19/77
ENG. Larry L. Horn 12/04/77
PROJ. ENG. [illegible] 12/19/77
PROD. [illegible] 12/19/77
NEXT HIGHER ASSY. B-DD-M8029-Ø
SCALE —
SHEET 1 OF 4

FIRST USED ON RX02
digital
TITLE RXV21 LSI11 (DI) CONTROLLER
SIZE D CODE CS NUMBER M8029-Ø-1 REV. C
DIST.

REV C | CODE CS | NUMBER M8029-Ø-1 | SIZE D

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT ©1977 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
RXV21 LSI11 CONTROLLER (D21)	D	CS	M3029-0-1	C
SCALE — SHEET 2 OF 4	DIST.			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT ©1977 DIGITAL EQUIPMENT CORPORATION"

R8 680 R7 330 +5V

D3 EN DATA L — 7404 E23 — D3 EN ADDR H — 74LS02 E3

D3 RD DB L — 7408 E16 — 8640 E14 — D3 XMIT EN 2 H

+5V R5 330 — D3 NPR REQST (1) H — DC0110 E7 REQ — BOMGI L — AR2 — R6 680

D4 OUT H — DATIO

+3V2 — CNTEN

D3 SYNC H — BUSYNC

D3 MAS RPLY H — RPLY

TMOUT

D3 8.2 MHZ — CLK

D2 INIT H — E11 — 74LS02 — INIT

BDMR — D3 B DMR L

MASTER — D3 M REPLY H — 7408 E16 — D3 MAS RPLY H

BDMGO — D3 B DMG O L

SYNC — D3 SYNC O H

DIN — D3 M DOUT H

DOUT — D3 M DIN H — E19 — 74LS04 — D3 M DIN L

ADREN — D3 EN ADDR H

DATEN — D3 EN DATA L

8881 E20 — BNI — D3 B SACK L

8881 E20 — D4 OUT H — AK2 — D3 BWT BT L

+3V1 — 74LS74 E12 — D3 SYNC Ø H — D3 M REPLY H — 74LS02 E11 — D2 INIT H

D3 RD CSR L — 7408 E16 — 8640 E14 — D3 YMIT EN 1 H

D4 ENA 12 BIT L — 7408 E16 — D3 DAL REC H

C7 1000pf — R2 39.2K 1% +5V — 74123 E6 — D2 INIT L

+3V1 — 74LS74 E12 — D3 NXM (Ø) H

+3V1 — 74LS74 E10 — D3 NPR REQST (1) H — D4 NPR REQ H

D3 NXM (1) H

+5V — 8.2MHZ OSC E8 — W1 — D3 8.2 MHZ

B RPLY — AF2 — D3 SRPLY H — D3 REPLY H

B DIN — AH2 — D3 DIN H — 74LS04 E15 — D3 DIN L

B DOUT — AE2 — D3 D OUT H — 74LS04 E15 — D3 D OUT L

B SYNC — AJ2 — D3 SYNC H — 74LS04 E15 — D3 SYNC L

8641 E1

D3 SEL CS L — 74LS02 E21 — 7404 E23 — D3 WRT CSR L

D3 PROC OUT L — D3 WRT CSR H

D3 WRT CSR PLS H

SRPLY L — E21 — 74LS02 — 7404 E23 — D3 WRT CSR PLS L

D3 SEL DB L — E21 — 74LS02 — E15 — 74LS04 — D3 WRT DB L — D3 WRT DB H

+5V — R10 330 — AM2 — R11 680

D4 B DONE H — DC0003 E26 RQSTA — BIAKI — BINIT — SDIN — RQSTB — ENA DAT — ENB DAT — ENA CLK — ENB CLK

D2 INIT L — D3 DIN L — DI TS BUS 8 H

ENA ST — BIRQ — AL2 — D3 IE H — D3 BIRQ L

BIAKO — INITO — VECTOR — D3 ENB VEC H

VEC RQST — ENB ST

+5V — R13 330 — AN2 — R12 680 — D3 BIAKOL

D3 WRT CSR PLS H

D3 PROC OUT L — SRPLY L — E2 — 74LS32 — E2 — 74LS32 — D4 SHIFT L — 74LS10 E17 — D3 DB CLK H — D2 MATCH H

D4 D4 LD BAR H

D3 M DIN L — E9 — 74LS00 — R15 1K 1% — C21 120 pf 5%

C9 220pf 1% — +5V — R4 4.64K 5%

74LS04 E15 — D3 SRPLY L — D3 SRPLY H — R3 — +5V

D4 ENA 12 BIT H — DI TS BUS 1 H — +3V2 — D3 ENB VEC H

DC0004 E27 — ENB — RXCX — BRPLY — SYNC — DAL 2 — DAL 1 — DAL Ø — WTBT — DOUT — DIN — VECTOR — SEL 6 — SEL 4 — SEL 2 — SEL Ø — OUT HB — OUT LB — INWD

74LS32 E28 — D3 RD DB L

74LS32 E28 — D3 RD CSR L

D3 IN WRD L

D3 SEL CS L

D3 SEL DB L

D3 PROC OUT L

REVISIONS

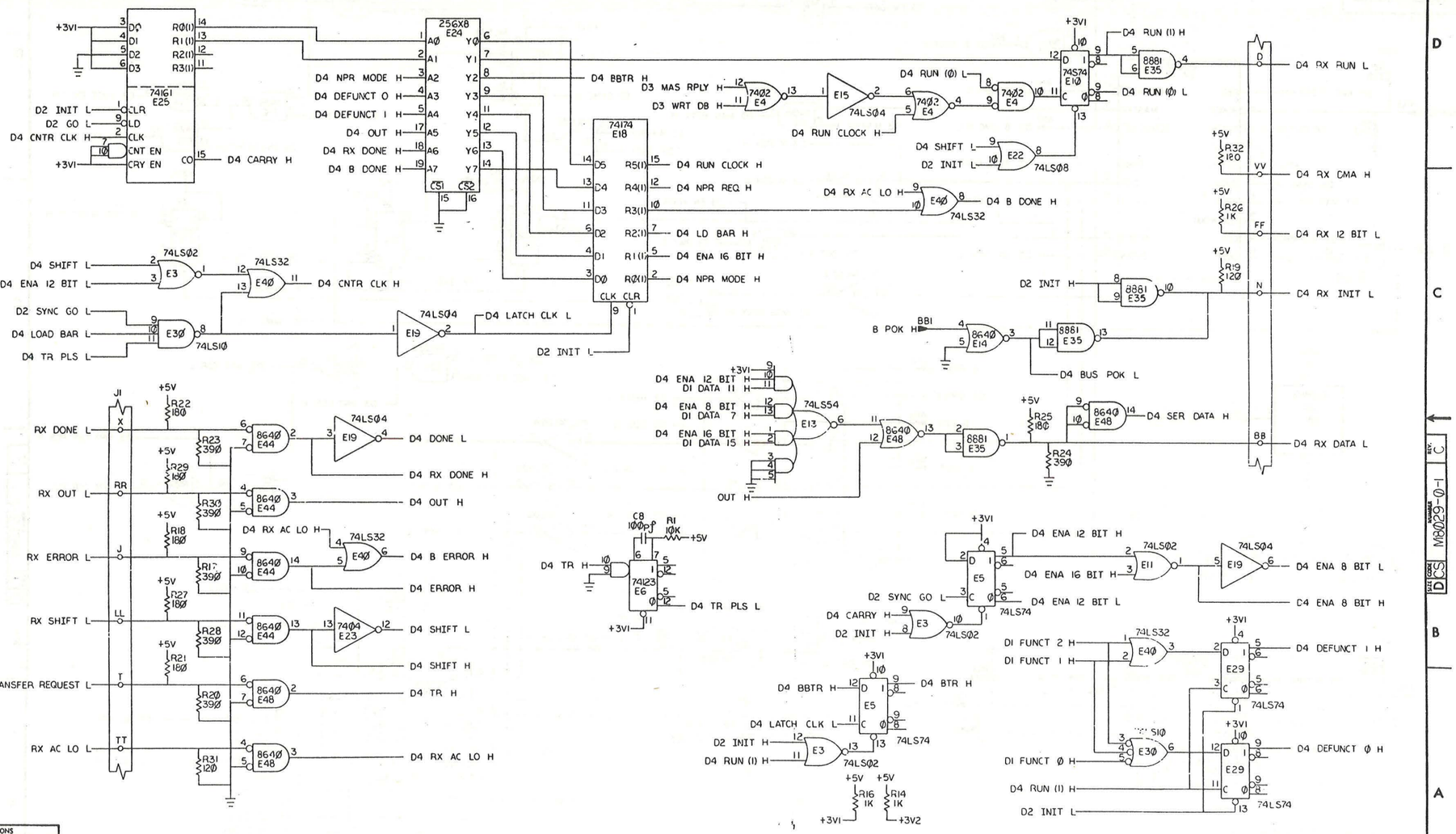
CHK	CHANGE NO.	REV.

TITLE		SIZE	CODE	NUMBER	REV.
RXV21 LSI11 CONTROLLER (D3)		D	CS	M8029-0-1	C
SCALE	SHEET 3 OF 4	DIST.			

DEC FORM NO. ORD 128

43

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT ©1977 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

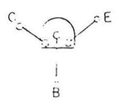
TITLE		SIZE	CODE	NUMBER	REV.
RXV21 LSI11 CONTROLLER	(D4)	D	CS	M8029-Ø-1	C
SCALE —	SHEET 4 OF 4	DIST.			

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1975 DIGITAL EQUIPMENT CORPORATION"

NOTES:

1. THIS MODULE PLUG IS COMPATABLE WITH:
   A. BC05M TTY CABLE.
   B. BC05C MODEM CABLE.
2. THE POLARITY REVERSES WHEN USING THE 20 MA CURRET LOOP TRANSMITTER OR RECEIVER IN THE PASSIVE CONFIGURATION. WHEN USING THE BC05M CARE MUST BE TAKEN TO SWAP THE APPROPRIATE WIRES.

3. A. INSERT WITH ACCORDANCE TO VENDOR PACKAGE.
VENDOR NAME:
G.E.34(GE) MOTOROLA, ITT, NATIONAL TRANSISTOR, NATIONAL SEMICONDUCTOR.



TOP VIEW

E.G.P.34 (VENDOR E) IS THE ONLY VENDOR PACKAGE MOUNTED WITH THE FLAT REVERSED.

4. PARTS IN THE FOLLOWING ALTERNATE PARTS LIST MAY BE USED IN PLACE OF THE CORRESPONDING PRIME PART WHEN NECESSARY.

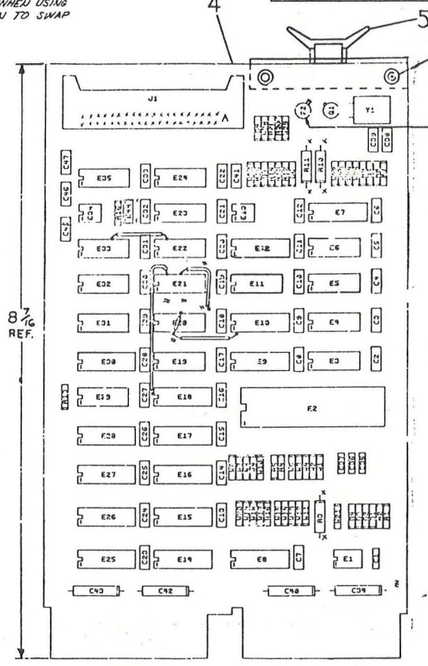
QTY	REF DESIGNATION	DESCRIPTION	PART NO.	ITEM NO.
31	C2 THRU 25, 27 THRU 33	CAPACITOR .01 uf 100V +80-20%	1001610-01	8A

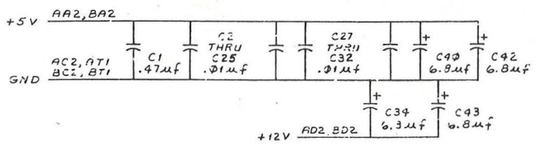
5. USE ADHESIVE (ITEM 60) WHERE NECESSARY TO SECURE REWORK WIRES.
6. THE FOLLOWING JUMPER REFERENCE DESIGNATIONS ARE NOT USED: W26, W27 (REPLACED BY RESISTORS R20, R21 PER ECO # M7940-00002).
7. THE FOLLOWING JUMPERS MAY BE INSERTED BY MANUFACTURING IN ORDER TO TEST AT 9600 BAUD, BUT WILL BE REMOVED BEFORE FINAL TEST AT 110 BAUD AND SHIPMENT TO CUSTOMER: W2, W22 THRU W24, W30.

I.C. DEC MHD26C	3			0
I.C. DEC HSUART	3	1	2	
I.C. DEC 1488			1	14
I.C. DEC 7493	10	5		
I.C. DEC 74175	8	16		
I.C. DEC 74157	8	16		
I.C. DEC 74155	8	16		
I.C. DEC 8837	8	16		
I.C. DEC 8641	8	16		
I.C. DEC 8136	5	10		
IC TYPE	GND	+5V	-12V	+12V

GND AND 5V ARE USUALLY PIN 7 AND 14 RESPECTIVELY EXCEPTIONS ARE STATED ABOVE

IC PIN LOCATIONS





8	W6, W7, W10, W16, W20, W21, W28, W29,	INSULATED JUMPER WIRE	9009185	55
1		HANDLE, FLIP CHIP, MAGENTA	9008337-8	56
2		EYELET	9008732	57
6		WIRE #30 AWG BUSS (RETROFIT)	9105740-55	58
2	R20, R21	RESISTOR, 180 Ω, 1/4W 5%	1301322	59
A/R		ADHESIVE	4901230	60

REF		X-Y COORDINATE HOLE LOCATION	K-CO M7940-0-4	1
REF		ASSY/DRILLING HOLE LAYOUT	D-AH-M7940-0-5	2
REF		MODULE ECO HISTORY	B-MH-M7940-0-8	3
1		ETCHED CIRCUIT BOARD	5011558	4
2	C38, C39	CAPACITOR, 33pf, 100V, 5%	1000009	5
4	C41, C45, C46, C47	CAPACITOR, 330pf, 100V, 5%	1000023	6
1	C29	CAPACITOR, 820pf, 100V, 5%	1000027	7
31	C2 THRU C25, C27 THRU C33	CAPACITOR, .01uf, 50V, +80 -20	1001610-00	8
1	C44	CAPACITOR, 5600 Pf, 100V, 20%	1001785	9
4	C34, C40, C42, C43	CAPACITOR, 6.8uf, 35V, 10%	1005308	10
4	C35, C36, C37, C1	CAPACITOR, .47uf, 25V, 20%	1010279	11
4	D1, D2, D3, D5	DIODE, D864	1100114	12
2	D6, D7	DIODE, 1N4004	1105798	13
1	D4	DIODE, 1N759A	1110828	14
1	D8	DIODE, MCL1301	1105810	15
1	J1	CONNECTOR, 40 PIN, R.ANG. HEADER 3M	1209941-02	16
2	R4, R5	RESISTOR, 47 OHM, 1/4W, 5%	1300202	17
1	R13	RESISTOR, 68 OHM, 1/4W, 5%	1300219	18
1	R3	RESISTOR, 120 OHM, 1/2W, 5%	1300243	19
1	R15	RESISTOR, 150 OHM, 1/4W, 5%	1300250	20
2	R7, R14	RESISTOR, 220 OHM, 1/4W, 5%	1300271	21
1	R8	RESISTOR, 330 OHM, 1/4W, 5%	1300295	22
2	R10, R11	RESISTOR, 270 OHM, 1/2W, 5%	1300285	23
2	R1, R2	RESISTOR, 470 OHM, 1/4W, 5%	1300316	24
1	R16	RESISTOR 2.4K OHM, 1/4W, 5%	1303177	25
1	R12	RESISTOR, 68K OHM, 1/4W, 5%	1301327	26
1	R19	RESISTOR, 100 OHM, 1/4W, 5%	1300229	27
1	R9	RESISTOR, 1GM, 1/4W	1302666	28
2	E14, E28	RESISTOR NETWORK, 330/680 OHM	1311003-02	29
1	Q1	TRANSISTOR, DEC 8534D	1502409	30
1	Q2	TRANSISTOR, MXA55	1510705	31
1	Y1	XTAL, 2.4576 MHZ	1812398	32
3	E5, E31, E33	I.C. DEC 7474	1905547	33
1	E22	I.C. DEC 7410	1905578	34
1	E11	I.C. DEC 7430	1905578	35
1	E23	I.C. DEC 7402	1909054	36
1	E6	I.C. DEC 7493	1909054	37
1	E4	I.C. DEC 74S04	1910534	38
3	E3, E19, E29	I.C. DEC 8881	1909705	39
2	E21, E32	I.C. DEC 7408	1910155	40
1	E24	I.C. DEC 1488	1910322	41
1	E35	I.C. DEC 1489	1910323	42
1	E20	I.C. DEC 74S74	1910544	43
1	E9	I.C. DEC 74175	1910651	44
3	E17, E27, E30	I.C. DEC 74157	1910655	45
3	E8, E12, E25	I.C. DEC 8837	1911116	46
2	E15, E18	I.C. DEC 8641	1911579	47
2	E13, E34	I.C. DEC 4N26-01	1911998-01	48
1	E1	I.C. DEC MH0026C	1912098	49
2	E16, E26	I.C. DEC 8136	1912395	50
1	E10	I.C. DEC 74155	1910656	51
1	E7	I.C. DEC 34702	2112622-00	52
1	E2	I.C. DEC HSUART (40 PIN)	2111450	53
2	TP1, TP2	SPLIT LUG	9008735	54

QTY	REF. DESIGNATION	DESCRIPTION	PART NO.	ITEM NO.

FIRST USED ON OPTION MODEL: LSI 11

ETCH BOARD REV: B

PARTS LIST

REV.	REVISIONS	CHARGE NO.	DATE	APPROVED
B	ECO M7940-00001		7-8-75	J. CYR
C	ECO M7940-00002		9-21-76	J. CYR
D	ECO M7940-00003		10-5-76	J. CYR
E	ECO M7940-00004		6-7-76	J. CYR
F	ECO M7940-00005		3-MAY-77	M. MICHAUD

DRN. J. Vincent DATE 4-2-75
CHKD DATE
ENG. DATE
PROJ. ENG. DATE
MFGR DATE

NEXT HIGHER ASSY

SCALE
SHEET 1 OF 5

TITLE: SERIAL LINE UNIT

digital

SIZE D | CODE CS | NUMBER M7940-0-1 | REV. F

DIST.

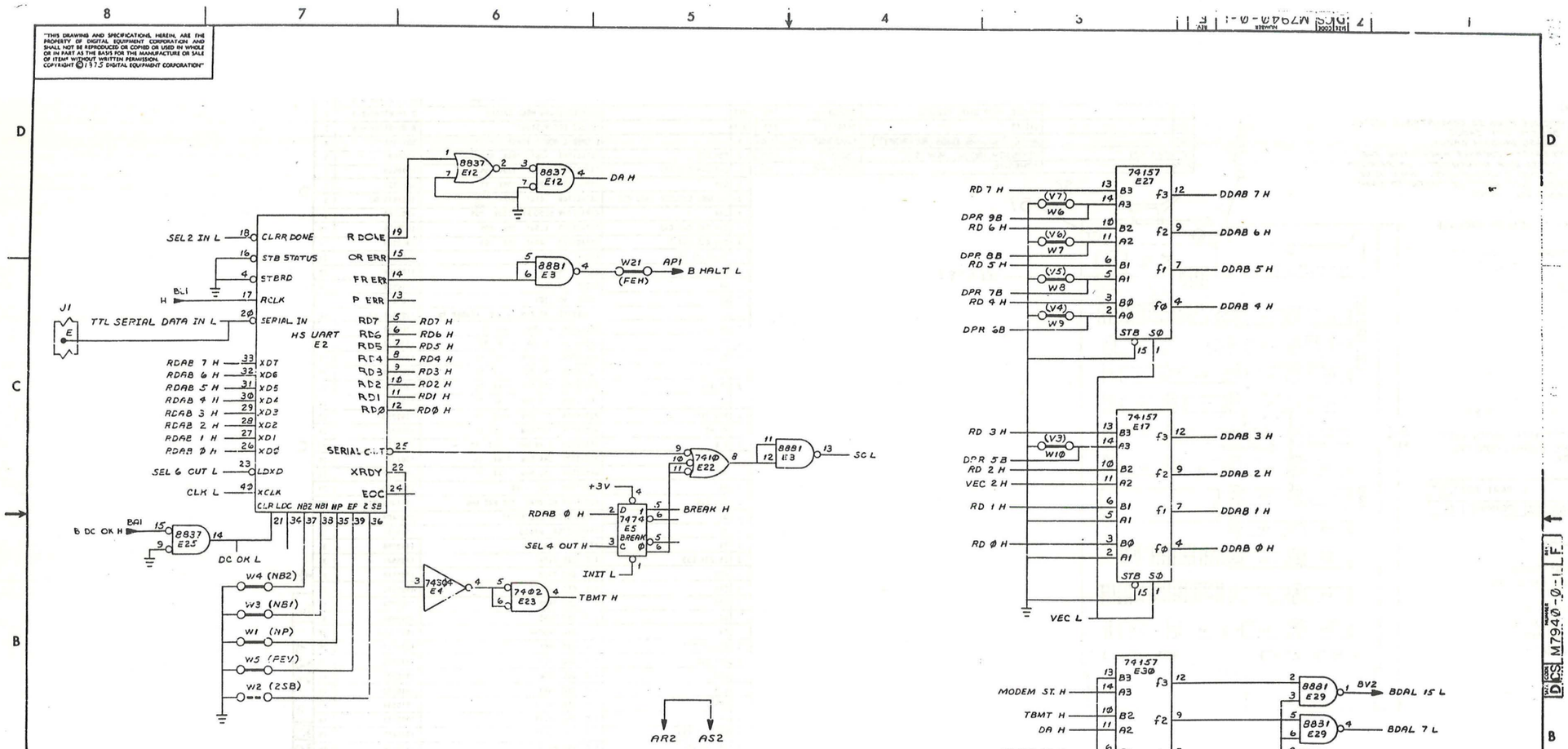
DEC NO. | EIA NO. | DEC NO. | EIA NO.

SEMICONDUCTOR CONVERSION CHART

DEC FORM NO. 040 135.8

45

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEM WITHOUT WRITTEN PERMISSION.
COPYRIGHT ©1975 DIGITAL EQUIPMENT CORPORATION"



(UART & MULTIPLEXOR LOGIC)

REVISIONS		
CHK	CHANGE NO.	REV.

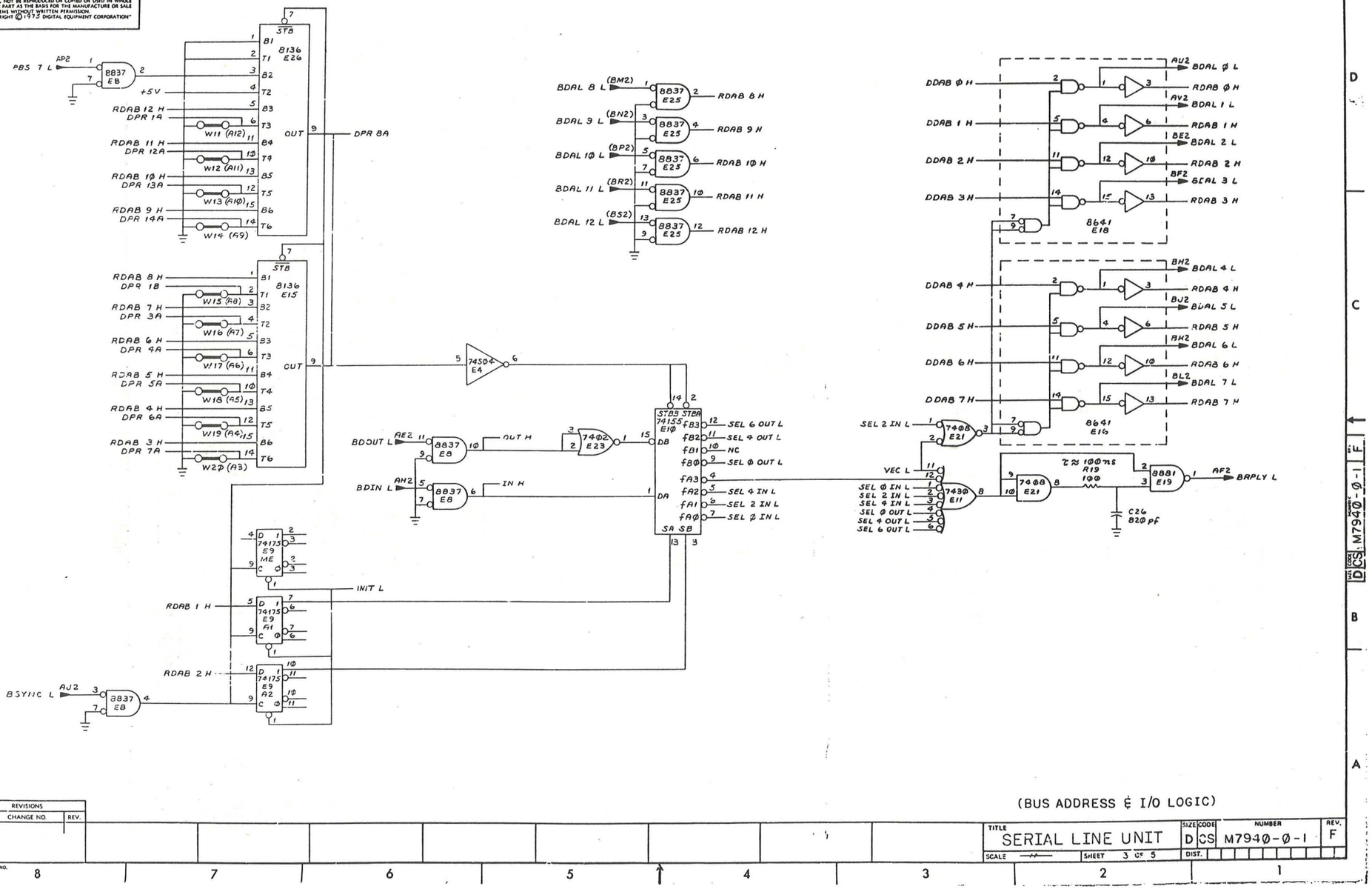
TITLE	SIZE	CODE	NUMBER	REV.
SERIAL LINE UNIT	D	CS	M7940-0-1	F
SCALE —/— SHEET 2 OF 5	DIST.			

DEC FORM NO. DRD 138

M7940-0-1 F

46

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1973 DIGITAL EQUIPMENT CORPORATION"



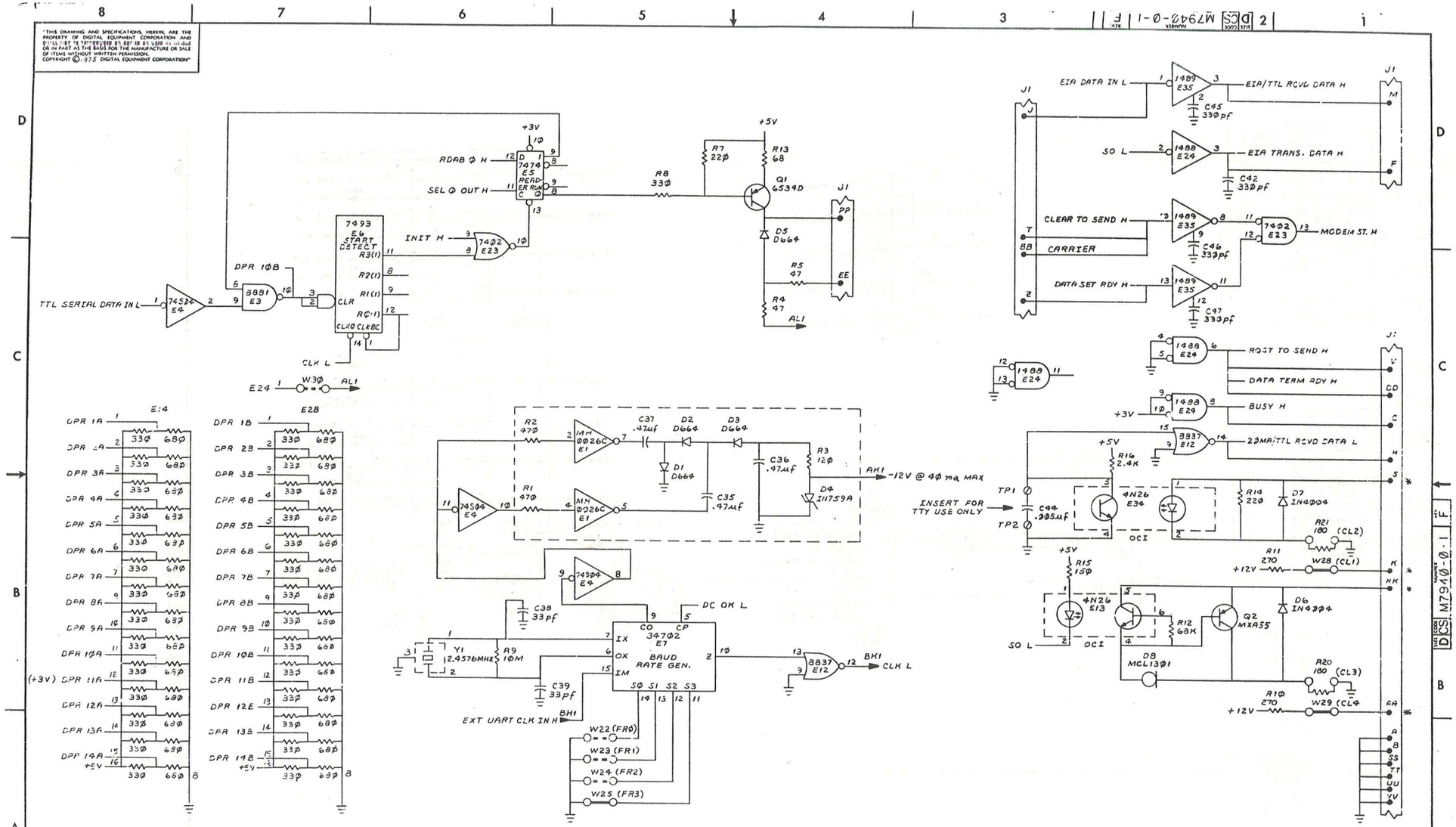
(BUS ADDRESS & I/O LOGIC)

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
SERIAL LINE UNIT	D	CS	M7940-0-1	F
SCALE —	SHEET 3 OF 5	DIST.		

DEC FORM NO. DRO 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR USED IN PART OR IN WHOLE AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1975 DIGITAL EQUIPMENT CORPORATION"



(PERIPHERAL & BAUD RATE LOGIC)

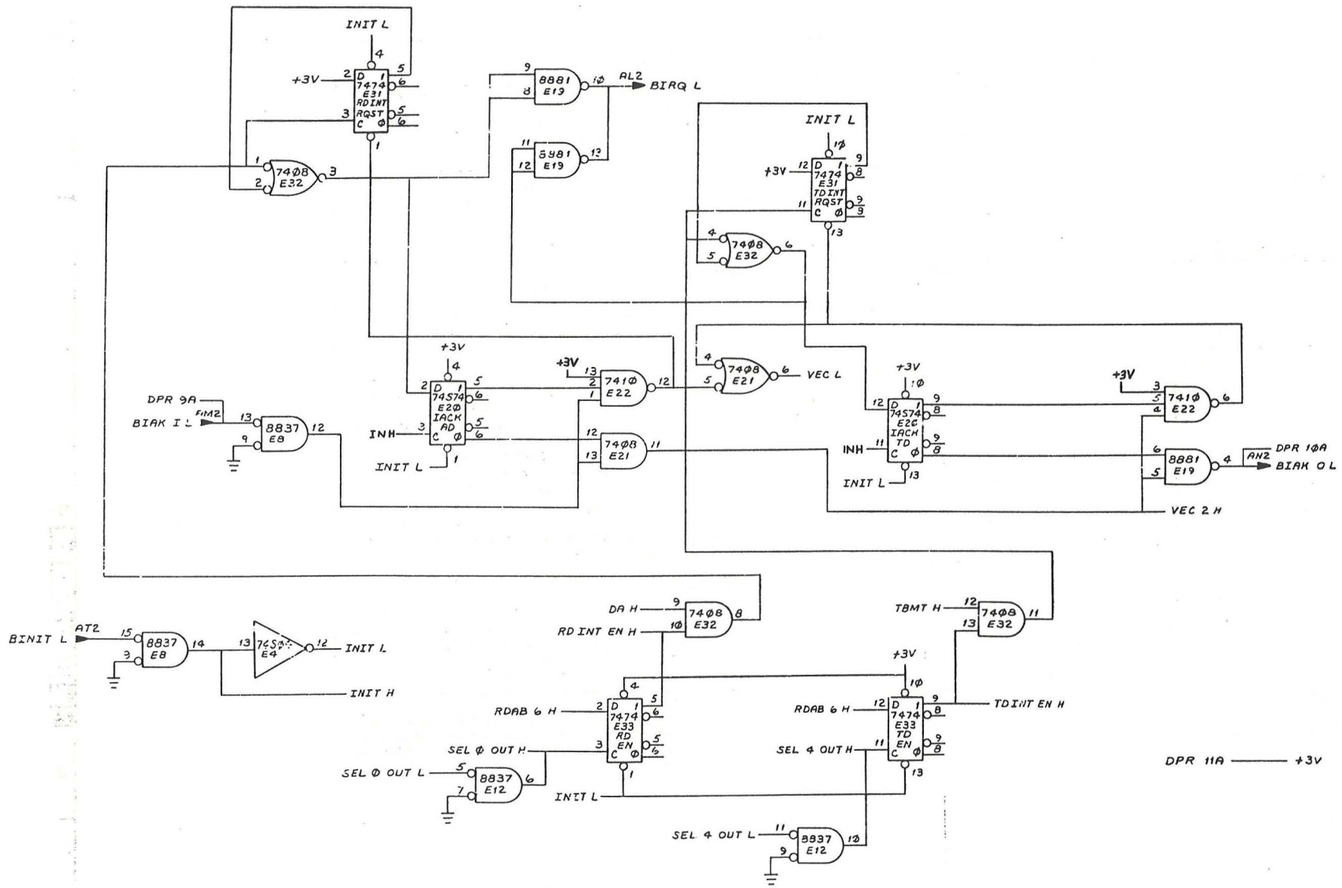
REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
SERIAL LINE UNIT	D	DCS	M7940-Ø-1	F
SCALE —	SHEET 1 OF 5	DIST.		

DEC 1 CHM NO. DRO 138

48

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1975 DIGITAL EQUIPMENT CORPORATION"



(INTERRUPT & ENABLE LOGIC)

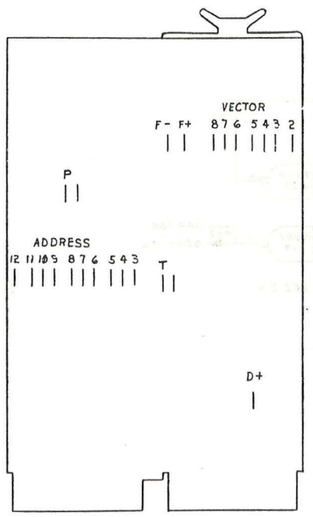
REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
SERIAL LINE UNIT	D	CS	M7940-0-1	F
SCALE —	SHEET 5 OF 5	DIST.		

DEC FORM NO. DND 138

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 , DIGITAL EQUIPMENT CORPORATION"

CROSS REFERENCE DRAWING
(SEE NOTE 3)



JUMPER TABLE		
FUNCTION	DESIGNATION	
ADDRESS SELECTION	A12-A3	DESIGNATIONS CORRESPOND TO THE BIT POSITION IN THE ADDRESS. HARD WIRE JUMPERS W2, W3 AND W4 BRIDGE A4, A5 AND A7 RESPECTIVELY, AND SPECIFY THE STANDARD ADDRESS OF: 177514 CS 177516 DB JUMPER IN = Ø JUMPER OUT = 1
VECTOR SELECTION	V8-V2	DESIGNATIONS CORRESPOND TO THE BIT POSITION IN THE VECTOR ALL VECTOR JUMPERS W9-W14 ARE INSERTED WITH THE EXCEPTION OF V7, TO SPECIFY THE STANDARD VECTOR OF 2ØØ. JUMPER IN = Ø JUMPER OUT = 1
CHARACTER TRANSLATION	W7	WHEN W7 IS INSERTED DATA FLOWS THROUGH LPV11 WITHOUT TRANSLATION. [STANDARD CONFIGURATION]
	T	WHEN T IS INSERTED DATA IS MODIFIED BY THE LPV11: LOWER CASE CHARACTERS ARE TRANSLATED TO UPPER CASE INSERT W7 OR T, DO NOT INSERT BOTH SIMULTANEOUSLY.
PARITY	W8	WHEN W8 IS INSERTED, P DATA 8 (LPØS PAPER INSTRUCTION / LA18Ø PARITY) WILL BE TRANSMITTED DIRECTLY [STANDARD CONFIGURATION]
	P	WHEN P IS INSERTED, P DATA 8 WILL BE HELD HIGH IF BOTH W8 AND P ARE REMOVED P DATA 8 WILL BE HELD LOW (USED WITH LPØS IF DAVFU IS PRESENT.) BOTH W8 AND P MUST NEVER BE INSERTED SIMULTANEOUSLY.
FAULT FILTER	F+	WHEN F+ (W6) IS INSERTED, NOISE IS FILTERED ON LPØS FAULT LINE. [STANDARD CONFIGURATION]
	F-	WHEN F- (W5) IS INSERTED, NOISE IS NOT FILTERED ON LPØS FAULT LINE. INSERT F+ OR F-, DO NOT INSERT BOTH SIMULTANEOUSLY.
DELAY	D	THIS JUMPER (W1) MUST BE INSERTED [STANDARD CONFIGURATION]. REMOVAL CAUSES BRPLY DELAY TO CHANGE, RESERVED FOR FUTURE USE

NOTE: 1. ALL DESIGNATIONS WHICH HAVE A W PREFIX ARE NOT LABELLED IN ETCH, ALL OTHERS ARE LABELLED IN ETCH (ADDRESS DESIGNATIONS SHOW ALTERNATELY, DUE TO A LACK OF SPACE).

NOTES:
1. TERMINATING RESISTORS MUST BE PRESENT ON THE CONNECTOR BOARD WITHIN THE LPØS.
2. IN THE EVENT THAT THE LPØS IS EQUIPPED WITH DAVFU, IT IS RECOMMENDED THAT THE PAPER INSTRUCTION (DATA 8) JUMPER W8 BE REMOVED, SINCE THE LPV11 DOES NOT SUPPORT THIS OPTION.
3. SEE UA DRAWING M8027-Ø-Ø FOR W DESIGNATIONS OF JUMPERS.

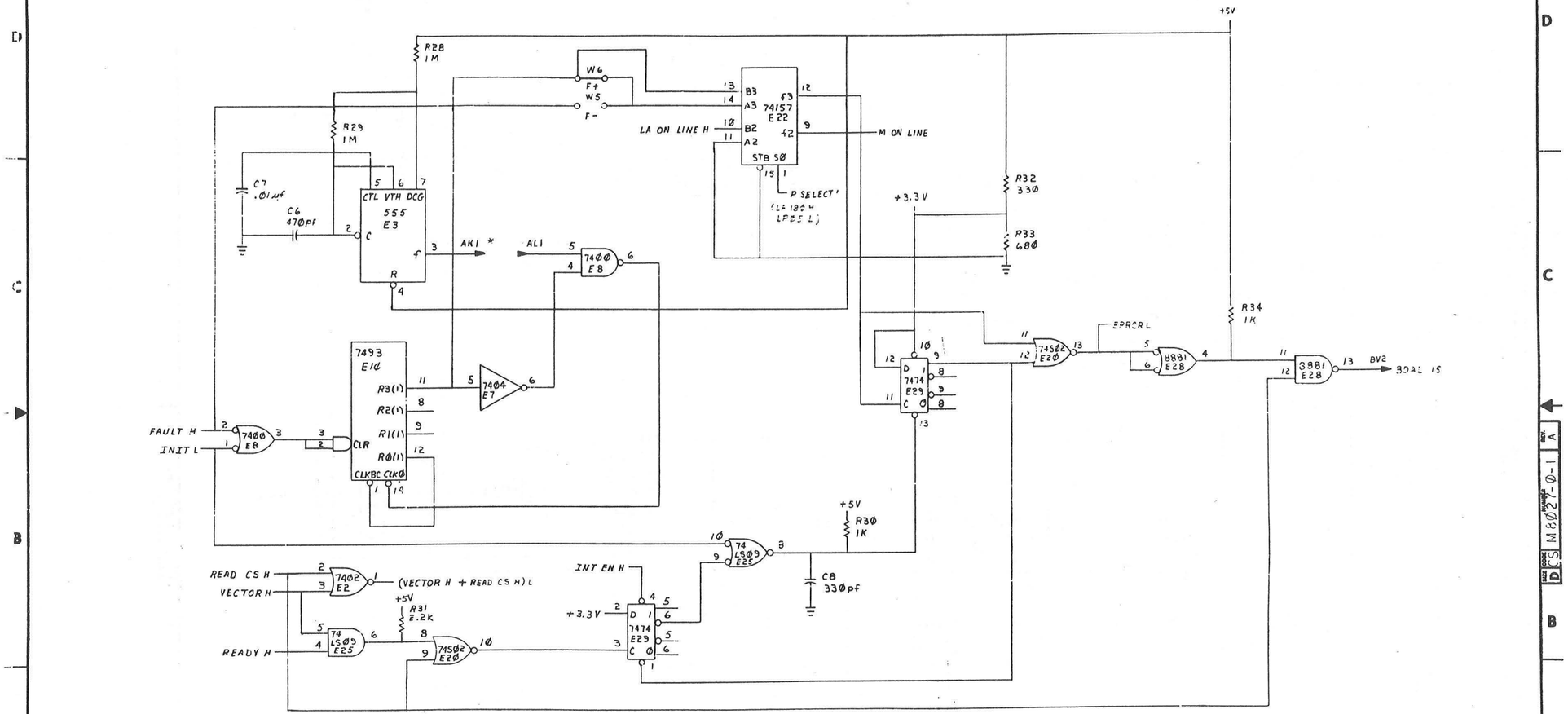
REVISIONS		
REV.	CHANGE NO.	CHK.

DRN. Silvia	7-28-77	FIRST USED ON
CHK'D		digital
ENG.		TITLE LPV11 PRINTER INTERFACE
PROJ. ENG.		
PROD.		
NEXT HIGHER ASSY.		
SCALE		SIZE D CODE CS NUMBER M8027-Ø-1 REV. A
SHEET 1 OF 5		DIST.

M8027-Ø-1 REV. A

50

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 , DIGITAL EQUIPMENT CORPORATION"



NOTE : * PINS AK1 & AL1 ARE JUMPERED THROUGH THE BACKPLANE.

DRN. Silvia	6-17-77	FIRST USED ON	digital
CHK'D Fly	20 Jan 77	TITLE	LPV-11 PRINTER INTERFACE
ENG. Fly	20 Jan 77		
PROJ. ENG. FP	20 Jan 77		
PROD.	2/7/77		
NEXT HIGHER ASSY.		SIZE D	CODE CS
SCALE		NUMBER M8027-Ø-1	REV. A
SHEET 2 OF 5		DIST.	

REVISIONS: REV. | CHANGE NO. | CHK.

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

J1
P DEMAND H — X
P BUSY L — W
P ON LINE H — F
P FAULT L — E
M CONN L — N
P CONN L — M
P SELECT (LA 18Ø L) (LP Ø5 H) — R, P, V, U, T, S

R1 330Ø
R2 330Ø
R3 150
R4 150
R5 150Ø
R6 150Ø
R7 1K
R8 1K
R9 4.02K
R26 1K
R27 2.2K
R36 150
R37 330
C1 .Ø1µf
C2 .Ø1µf
C3 18Øpf
C4 150 pf
C5 150pf
W1

8641 E16
7414 E21
7404 E7
8640 E12
74 LSØ3 E25
7474 E15
74 LSØ93 E25
7400 E8
8881 E28
74157 E22
DCØØ4 E1
74Ø2 E2

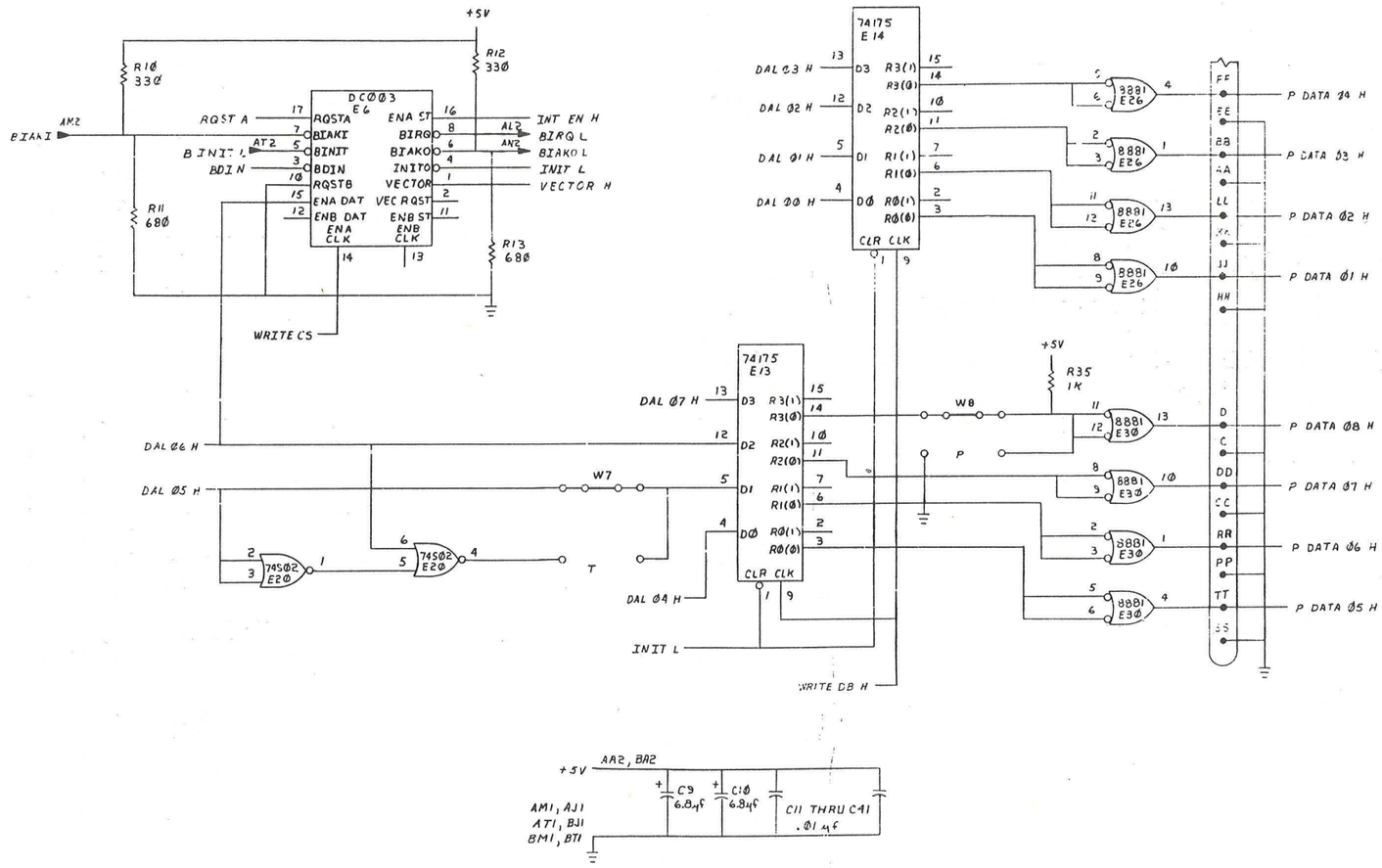
BUSY H
LA ON LINE H
FAULT H
P SELECT' (LA 18Ø H LPØ5 L)
LP STROBE L
READY · ERR
RQST A
ERROR L
INIT L
READY H
WRITE DB L
WRITE DB H
M CONN L
P STROBE
WRITE CS
READ CS H
BDIN
VECTOR H
BD IN L
BSYNC
DAL Ø2 H
DAL Ø1 H
DAL Ø Ø H
BWTBT
BDOUT
ENB H
BRPLY
AK2
AE2
AH2
AJ2
AF2
+3.3V
+5V

7493	1Ø	5
555	1	8
8641	8	16
8640	4	8
8136	8	16
74175	8	16
74157	8	16
DCØØ3	9	18
DCØØ4	1Ø	2Ø
IC TYPE	GND	+5

REVISIONS
CHK | CHANGE NO. | REV.

TITLE: LPV-11 PRINTER INTERFACE
SIZE: D | CODE: CS | NUMBER: MØØ27-Ø-1 | REV: A
SCALE: — | SHEET 3 OF 5 | DIST.

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"



REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
LPV-11 PRINTER INTERFACE	D	CS	M8027-0-1	A
SCALE —	SHEET 4 OF 5	DIST		

53

"THIS DRAWING AND SPECIFICATIONS, HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT © 1977 DIGITAL EQUIPMENT CORPORATION"

+5V

R39 1K
R38 1K
R43 1K
R42 1K
R41 1K
R40 1K
R44 1K

V10 V3
W9 V2
V7
READY-ERR
W13 V6
INT EN H
W12 V5
W11 V4
W14 V8

74157 E4
74157 E9

M ON LINE
BUSY H

(VECTOR H + READ CS H)L
VECTOR H
8881 E28
BM2 BDAL Ø8 L

8641 E18
8641 E23

AU2 BDAL ØØ L
DAL ØØ H
AV2 BDAL Ø1 L
DAL Ø1 H
BE2 BDAL Ø2 L
DAL Ø2 H
BF2 BDAL Ø3 L
DAL Ø3 H
BH2 BDAL Ø4 L
DAL Ø4
BJ2 BDAL Ø5 L
DAL Ø5 H
BK2 BDAL Ø6 L
DAL Ø6 H
BL2 BDAL Ø7 L
DAL Ø7 H

BDAL Ø8 BM2 8640 E12
BDAL Ø9 BN2 8640 E27
BDAL 10 BP2 8640 E27
BDAL 11 BR2 8640 E27
BDAL 12 BS2 8640 E27
BBS 7 AP2 8640 E12

BDMG1 L AR2 AS2 BDMG0 L

+5V
R19 1K R18 1K R17 1K R16 1K R15 1K R14 1K
A7 W4 A6 A5 W3 A4 W2 A3

R25 1K R24 1K R23 1K R22 1K R21 1K R20 1K
A12 A11 A10 A9 A8

8136 E13
8136 E24
OUT
ENE H
STB

REVISIONS		
CHK	CHANGE NO.	REV.

TITLE	SIZE	CODE	NUMBER	REV.
LPV-11 PRINTER INTERFACE	D	CS	M8027-0-1	A.
SCALE	SHEET 5 OF 5	DIST.		

54

TABLE OF CONTENTS

2-	1	BASIC DEFINITIONS
3-	1	LOCAL DATA
4-	1	SPECIAL EQUALITIES FOR BVD11-AA
5-	1	BVD11-AA SWITCH SETTINGS.
6-	1	DIAGNOSTIC LIGHT DISPLAY
7-	1	LIST OF ERROR HALTS
8-	1	START OF BVD11-AA ROM DIAGNOSTIC.
9-	1	CPU ERROR HALT ONE.
9-	20	LOOP SWITCH ROUTINE
10-	1	PRE TEST
11-	75	PRE TEST 1E
11-	139	PRE TEST 1F
12-	1	SWITCH CHECK
14-	1	RK-11 BOOT
15-	1	CHANGE PAGE ROUTINE FOR PAGE ONE.

REV	CHK			
		FIRST USED ON OPTION MODEL		DIGITAL EQUIPMENT CORPORATION MAYNARD, MASSACHUSETTS
		DRN. GOODRICH	DATE 25 OCT 77	TITLE BDV11-AA DIAGNOSTIC/BOOT ROM LISTING
		CHK'D.	DATE 25OCT77	
		ENG.	DATE 10/25/77	
		PROJ.ENG.	DATE	
		PROJ. K Hamilton	DATE 10/11/77	SIZE K / CODE CS / NUMBER BDV11-A-2 / REV
	SHEET 1 OF 118	NEXT HIGHER ASSEMBLY B-DD-BDV11-A		DIST.

"THIS DRAWING AND SPECIFICATIONS HEREIN, ARE THE PROPERTY OF DIGITAL EQUIPMENT CORPORATION AND SHALL NOT BE REPRODUCED OR COPIED OR USED IN WHOLE OR IN PART AS THE BASIS FOR THE MANUFACTURE OR SALE OF ITEMS WITHOUT WRITTEN PERMISSION.
COPYRIGHT 1977, DIGITAL EQUIPMENT CORPORATION"

```
                        .SBTTL  BASIC DEFINITIONS

                        ;*INITIAL ADDRESS OF THE STACK POINTER *** 177 524 ***
177524                  STACK=  177524

                        ;*MISCELLANEOUS DEFINITIONS
000012                  LF=:    12              ;;CODE FOR LINE FEED
000015                  CR=     15              ;;CODE FOR CARRIAGE RETURN

                        ;*GENERAL PURPOSE REGISTER DEFINITIONS
000000                  R0=     %0              ;;GENERAL REGISTER
000001                  R1=     %1              ;;GENERAL REGISTER
000002                  R2=     %2              ;;GENERAL REGISTER
000003                  R3=     %3              ;;GENERAL REGISTER
000004                  R4=     %4              ;;GENERAL REGISTER
000005                  R5=     %5              ;;GENERAL REGISTER
000006                  R6=     %6              ;;GENERAL REGISTER
000007                  R7=     %7              ;;GENERAL REGISTER
000006                  SP=     %6              ;;STACK POINTER
000007                  PC=     %7              ;;PROGRAM COUNTER

                        ;*DATA BIT DEFINITIONS (BIT00 TO BIT15)
100000                  BIT15=  100000
040000                  BIT14=  40000
020000                  BIT13=  20000
010000                  BIT12=  10000
004000                  BIT11=  4000
002000                  BIT10=  2000
001000                  BIT9=   1000
000400                  BIT8=   400
000200                  BIT7=   200
000100                  BIT6=   100
000040                  BIT5=   40
000020                  BIT4=   20
000010                  BIT3=   10
000004                  BIT2=   4
000002                  BIT1=   2
000001                  BIT0=   1
```

LOCAL DATA

```
                                .SBTTL  LOCAL DATA
                        
        177520          PAGELO  =       177520          ; PAGE CONTROL REGISTER
                        
        177521          PAGEHI  =       177521                  ; LOW PAGE REGISTER = BITS 0-7
                                                                ; HIGH PAGE REGISTER=   BITS 8-15
                                                                ; THIS IS A READ/WRITE ONLY REGISTER
                        
        177522          RWADD   =       177522          ; READ/WRITE 16 BIT REGISTER
                                                                ; THIS IS A READ WRITE LOCATION
                                                                ; USED IN THE CPU TESTS
                                                                ; THIS LOC. IS BYTE/WORD ADDRESSABLE
                        
        177524          SWITCH  =       177524          ; TERMINATOR SWITCH ADDRESS
                                                                ; THIS IS WHERE TO READ THE
                                                                ; SWITCH SETTINGS FOR THE
                                                                ; TERMINATOR BOARD. BITS 0-11
                                                                ; THIS IS A READ ONLY REGISTER
                        
        177524          LIGHTS  =       177524          ; THIS IS THE REGISTER TO CONTROL
                                                                ; THE DIAGNOSTIC LIGHT DISPLAY
                                                                ; BITS USED ARE 00 - 03
                                                                ; IF A BIT=1 THE CORESPONDING LED IS OFF
                                                                ; IF A BIT= 0 THE CORESPONDING LED IS ON
                                                                ; THIS IS  A WRITE ONLY REGISTER
                        
        177170          RXCS    =       177170          ; FLOPPY CONTROL AND STATUS REGISTER
                                                                ; FLOPPY DATA BUFFER REGISTER
        177172          RXDB    =       177172
                        
                                                        ; CONSOL  KEYBOARD ADDRESS
        177560          TKS     =       177560                  ; KEYBOARD STATUS REGISTER
        177562          TKB     =       177562                  ; KEYBOARD DATA BUFFER
                        
                                                        ; CONSOL DISPLAY ADDRESS
        177564          TPS     =       177564                  ; DISPLAY STATUS REGISTER
        177566          TPB     =       177566                  ; DISPLAY DATA BUFFER
                        
                        
        000226          SSYNC   =       226                     ; SYNC CHARACTER
        000377          ASYN    =       377                     ; ASYNC CHARACTERS
        000220          DLE     =       220                     ; DDCMP DLE CHARACTER
        000005          ENQ     =       005                     ; DDCMP ENQ CHARACTER
        000201          SOH     =       201                     ; DDCMP SOH CHARACTER
        120001          POLY    =       120001                  ; CONSTANT FOR CRC-16
```

SPECIAL EQUALITIES FOR BVD11-AA

```
                                .SBTTL  SPECIAL EQUALITIES FOR BVD11-AA
                                ;* DIAGNOSTIC LIGHT DISPLAY VALUES
000016                          ONE     =BIT3 ! BIT2 ! BIT1             ; 1     THE BITS THAT ARE OFF (0)
000015                          TWO     =BIT3 ! BIT2 ! BIT0             ; 2     INDICATE WHICH LED IS ON.
000014                          THREE   =BIT3 ! BIT2                    ; 3
000013                          FOUR    =BIT3 ! BIT1 ! BIT0             ; 4
000012                          FIVE    =BIT3 ! BIT1                    ; 5
000011                          SIX     =BIT3 ! BIT0                    ; 6
000010                          SEVEN   =BIT3                           ; 7
000007                          TEN     =BIT2 ! BIT1 ! BIT0             ; 10
000006                          ELEVEN  =BIT2 ! BIT1                    ; 11
000005                          TWELVE  =BIT2 ! BIT0                    ; 12
000003                          A3TEEN  =BIT1 ! BIT0                    ; 13
000003                          A4TEEN  =BIT1 ! BIT0                    ; 14
000002                          A5TEEN  =BIT1                           ; 15
000001                          A6TEEN  =BIT0                           ; 16
000037                          BLANK   =BIT4 !BIT3 ! BIT2 ! BIT1 ! BIT0
                                ;* PAGE CONTROL REGISTER SELECTIONS
000400                          PAGE0   =       000400          ; ROMA - PRE-TEST,SWITCH CHECK,LOOP TEST,RKV BOOT.

001402                          PAGE2   =       001402          ; ROMB - MEMORY TEST AND CPU TEST.

002404                          PAGE4   =       002404          ; ROMC - CPU TEST

003406                          PAGE6   =       003406          ; ROMD - CONSOL TERMINAL TEST.

004410                          PAGE10  =       004410          ; ROME - RLV01 BOOTSTRAP.

005412                          PAGE12  =       005412          ; ROMF - FLOPPY (RX01 - RX02) BOOTSTRAP.

006414                          PAGE14  =       006414          ; ROMG - SLU BOOTSTRAPS.

007416                          PAGE16  =       007416          ; ROMH - ROM BOOTSTRAPS.

000020                          PAGE20  =       020             ; ROM ADDRESS FOR DIAGNOSTIC ROM 2.

000040                          PAGE40  =       040             ; ROM ADDRESS FOR EPROM.

000200                          PAG200  =       200             ; ADDRESS FOR SYSTEM ROM .
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS

173000                          SLBOOT  =       173000          ; SERIAL LINE BOOTSTRAP         PAGE 14
173000                          RLBOOT  =       173000          ; RLV-01 BOOTSTRAP              PAGE 10
173000                          ROBOOT  =       173000          ; ROM BOOTSTRAP                 PAGE 16
173000                          RXBOOT  =       173000          ; FLOPPY BOOTSTRAP              PAGE 12
173516                          TEST1   =       173516          ; CPU TESTS                     PAGE 2
173000                          MMTEST  =       173000          ; MEMORY TEST                   PAGE 2
173302                          VTTEST  =       173302          ; CONSOL TERMINAL TEST          PAGE 6
000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          . = 173000
```

BVD11-AA SWITCH SETTINGS.

```
                        .SBTTL  BVD11-AA SWITCH SETTINGS.
                        ;*                                      FLOPPY  ROM             SLU
                        ;* SWITCH POSITION A1    = CPU1 TEST
                        ;*
                        ;* SWITCH POSITION A2    = MEMORY TEST
                        ;*
                        ;* SWITCH POSITION A3    = SLU BOOT
                        ;*
                        ;* SWITCH POSITION A4    = CONSOL TERMINAL TEST                  DUV-11=1
                        ;*
                        ;* SWITCH POSITION A5    = ROM BOOT                              DLV-11= 1
                        ;*
                        ;* SWITCH POSITION A6    = FLOPPY BOOT                           DUP-11= 1
                        ;*
                        ;* SWITCH POSITION A7    = RL01 BOOT     RX01 = 0                OFF. FUTURE USE.
                        ;*                                       RX02 = 1
                        ;*
                        ;* SWITCH POSITION A8    = RK-11 BOOT
                        ;*
                        ;* SWITCH POSITION B1    = LOOP ON TEST          DIAG. ROM 2
                        ;*
                        ;* SWITCH POSITION B2                            PROM
                        ;*
                        ;* SWITCH POSITION B3                            SYS. ROM
                        ;*
                        ;* SWITCH POSITION B4                            B4=0 ROM IS 1K
                        ;* SWITCH POSITION B4                            B4=1 ROM IS 2K
```

DIAGNOSTIC LIGHT DISPLAY

```
                        .SBTTL  DIAGNOSTIC LIGHT DISPLAY
                        ;* DIAGNOSTIC LIGHT DISPLAY
                        ;* ------------------------
                        ;* 1  = CPU ERROR.
                        ;*
                        ;* 2  = MEMORY ERROR.
                        ;*
                        ;* 3  = CONSOL TERMINAL TEST DISPLAY ERROR.
                        ;*
                        ;* 4  = CONSOL TERMINAL TEST KEYBOARD ERROR.
                        ;*
                        ;* 5  = LOAD DEVICE STATUS ERROR.
                        ;*
                        ;* 6  = BOOTSTRAP CODE INCORRECT. (NOP INSTRUCTION NOT IN LOCATION 000000)
                        ;*
                        ;* 7  = SLU ERROR NO RESPONCE FROM HOST.
                        ;*
                        ;* 10 = SLU RECEIVED DONE FLAG SET.
                        ;*
                        ;* 11 = SLU MESSAGE RECEIVED IN DDCMP FORMAT.
                        ;*
                        ;* 12 = ROM BOOTSTRAP ERROR.
                        ;*
                        ;* 13 =
                        ;*
                        ;* 14 =
                        ;*
                        ;* 15 =
                        ;*
                        ;* 16 =
                        ;*
                        ;* 17 = HALT SWITCH ON OR UNABLE TO RUN.
```

DIGITAL EQUIPMENT CORPORATION MAYNARD, MASSACHUSETTS	TITLE	BDV11-AA DIAGNOSTIC/BOOT ROM LISTING	SHEET 6 OF 18	SIZE	CODE	NUMBER	REV
				K	CS	BDV11-A-2	

LIST OF ERROR HALTS

```
                        .SBTTL  LIST OF ERROR HALTS
                        ;
                        ; ADDRESS OF ERROR      PAGE            CAUSE OF ERROR
                        ; ----------------      ----            --------------
                        ;
                        ; 173 042       ROMG    6414            SLU SWITCH SELECTION INCORRECT
                        ;                                       ERROR IN SWITCHES B1, B2, B3, B4.
                        ; 173 046       ROMG    6414            SLU ERROR. CSR ADDRESS FOR SELECTED DEVICE IN ERROR.
                        ;                                       CHECK CSR FOR SELECTED DEVICE IN FLOATING CSR ADDRES
                        ;
                        ; 173 050       ROMA    400             CP1ERR, R0 CONTAINS ADDRESS OF ERROR.
                        ;
                        ; 173 070       ROMB    1402            MEMORY ERROR 1. WRITE ADDRESS INTO ITSELF.
                        ;
                        ; 173 120       ROMB    1402            MEMORY ERROR 2. DATA TEST FAILED.
                        ;
                        ; 173 154       ROMB    1402            MEMORY ERROR 3. WRITE AND READ BYTES FAILED.
                        ;
                        ; 173 220       ROMH    7416            ROM LOADER ERROR. CHECKSUM ON DATA BLOCK.
                        ;
                        ; 173 240       ROMD    3406            CP4ERR, R0 CONTAINS ADDRESS OF ERROR.
                        ;
                        ; 173 370       ROMH    7416            ROM LOADER ERROR. CHECKSUM ON ADDRESS BLOCK.
                        ;
                        ; 173 404       ROMH    7416            ROM LOADER ERROR. JUMP ADDRESS IS ODD.
                        ;
                        ; 173 522       ROMD    3406            IN CONSOL TERMINAL TEST A "NO" TYPED.
                        ;
                        ; 173 522       ROMF    5412            RX ERROR. R0 POINTS TO CAUSE OF ERROR.
                        ;
                        ; 173 504       ROME    4410            RL ERROR. R0 POINTS TO CAUSE OF ERROR.
                        ;
                        ; 173 550       ROMD    3406            CONSOL TERMINAL TEST. NO DONE FLAG.
                        ;
                        ; 173 634       ROMC    2404            CPU ERROR 3. R0 POINTS TO CAUSE OF ERROR.
                        ;
                        ; 173 734       ROMB    1402            CPU ERROR 2. R0 POINTS TO CAUSE OF ERROR.
```

```
                                .SBTTL  START OF BVD11-AA ROM DIAGNOSTIC.
                                ;*
                                ;;*************************************************************************
                                ;;*************************************************************************
                                ;* PRETEST
                                ;* START HERE WHEN THE INITIALIZE SWITCH IS ENABLED OR POWER IS APPLIED.
                                ;* SET DIAGNOSTIC LIGHT DISPLAY TO '1'.
                                ;* DO A "RESET".
                                ;* PSW = NO INTERRUPTS ALLOWED.
                                ;* SP  = RWADD+2
                                ;* SET PAGE CONTROL REGISTER TO PAGES 0 AND 1.
                                ;* MAKE ALL  REGISTERS (R1 TO R5) = ZERO (0).
                                ;* DO JSR TO FIRST TEST OF PRETEST.
                                ;* CP1 ERR LOCATED HERE FOR ALL ERRORS IN ROMA.
                                ;;*************************************************************************
                                ;;*************************************************************************

                                                        ; ROM PAGE IS PAGE0. 000400 IN PAGE CONTROL REGISTER.
                                                        ; DIAGNOSTIC LIGHT DISPLAY = 17
173000                  START:                          ; HARDWARE SETS LIGHTS TO 17 ON A INITILIZE OR POWER UP .
                                                        ; IF PC=173 000 AND LIGHTS=17 POSIBLE PROBLEMS CAN BE
                                                        ; HALT SWITCH SET
                                                        ; BUS ERROR
                                                        ; MEMORY LOCATION 4 NOT PRESENT OR BAD.

173000  112737  000016  177524          MOVB    #ONE,@#LIGHTS   ; FIRST INSTRUCTION SET LIGHT TO A '1'.
173006  000005                         RESET                   ; DIAGNOSTIC LIGHT DISPLAY = 1.
173010  012700  000340                 MOV     #340,R0         ; INITIALIZE CPU.

173014  106400                         MTPS    R0              ; CPU PRIORITY =  NO INTERRUPTS WILL OCCURE.

173016  012706  177524                 MOV     #RWADD+2,R6     ; SET STACK = 177524.

173022  012737  000400  177520         MOV     #PAGE0,@#PAGELO ; SET PAGE CONTROL REGISTER TO PAGE 0 AND 1.

173030  005001                         CLR     R1              ; CLEAR THE REGISTERS .
173032  005002                         CLR     R2              ; R0 = 340
173034  005003                         CLR     R3              ; R1 TO R5 = ZERO (0)
173036  005004                         CLR     R4              ; R6 = 177 524
173040  005005                         CLR     R5
173042  004067  000004                 JSR     R0,PTST1        ; TEST THE JSR INSTRUCTION
```

CPU ERROR HALT ONE.

```
                                .SBTTL  CPU ERROR HALT ONE.
                                        ; ***********************************************
                                        ; ERROR IN THE CPU PRE TESTS
                                                ; R0= ADDRESS OF FAILURE
                                                ; IF R0 = ADDRESS OF CP1ERR (177046)
173046  000000                  CP1ERR: HALT                  ; FAILURE IN PRETEST,TESTA-TESTB-TESTC
                                                ; OTHERWISE POINTS TO ROUTINE THAT FAILED.
                                                ; REFER TO ROUTINE IN ERROR FOR
                                                ; ERROR REGISTER VALUES.
                                                ; R6 = 177 524 OR 177 522.
                                                ; "ODT" PROCEED WILL RESTART THE TEST.
                                        ; ***********************************************
```

```
                                .SBTTL  LOOP SWITCH ROUTINE
                                ;;*******************************************************************************
                                ;;*******************************************************************************
                                ;* LOOP
                                ;* THIS INSTRUCTION WILL BE EXECUTED IF THE USER PROCEEDS FROM ERROR CP1ERR
                                ;* USING "ODT" OR THE LOOP SWITCH ON THE TERMINATOR BOARD WAS SET.
                                ;*
                                ;* LOOP ON TEST WILL RESTART THE ROM CODE AND CONTINUE TO RUN UNTIL A ERROR OR THE
                                ;* USER CHANGES THE SWITCH SETTING. NO BOOT SWITCH CAN BE SELECTED IF LOOP IS SET.
                                ;;*******************************************************************************
                                ;;*******************************************************************************

173050                          LOOP:
173050  000753                          BR      START           ; GO TO THE BEGINING OF THE ROM CODE.
```

```
                                .SBTTL  PRE TEST
                                ;;**************************************************************************
                                ;;**************************************************************************
                                ;* PTST 1 -  PTST 1A -  PTST 1B
                                ;*      SET THE CONDITION CODES (CC) AND TEST THE ABILITY OF THE
                                ;*      BRANCH INSTRUCTIONS TO EXECUTE CORRECTLY.
                                ;* PTST 1C
                                ;*      TEST THE RESULTS OF THE JSR INSTRUCTION.
                                ;* PTST 1D
                                ;*      TEST ALL SINGLE OPERAND INSTRUCTIONS
                                ;* PTST 1E
                                ;*      TEST ALL BYTE SINGLE OPERAND INSTRUCTIONS
                                ;* PTST 1F
                                ;*      TEST THAT THE INSTRUCTIONS USED IN THE SWITCH CHECKING
                                ;*      ROUTINE WILL EXECUTE CORRECTLY BEFORE ENTERING THAT ROUTINE.
                                ;;**************************************************************************
                                ;;**************************************************************************
                                        ; DIAGNOSTIC LIGHT DISPLAY = "1"
173052  000257          PTST1:  CCC                     ; CLEAR THE CONDITION CODES
                                                        ; Z=0   N=0     V=0     C=0     PSW=340
173054  001774                  BEQ     CP1ERR          ; CHECK ALL THE BRANCH INSTRUCTIONS
173056  100773                  BMI     CP1ERR          ; THAT WILL NOT BRANCH WITH THE
173060  102772                  BVS     CP1ERR          ; CONDITION CODES SET FOR PRE TEST 1.
173062  103771                  BCS     CP1ERR          ; ERROR WILL CAUSE A HALT AT CP1 ERR.
173064  002770                  BLT     CP1ERR          ; R0 = CP1ERR ADDRESS(177 046)
173066  003767                  BLE     CP1ERR          ; R1 TO R5 = ZERO (0)
173070  101766                  BLOS    CP1ERR          ; R6 = RWADD ADDRESS
173072  101015                  BHI     BRBACK          ; DO THIS BRANCH WHERE OTHERS DID NOT .

                                                        ; PRE TEST 1-A TEST THE BRANCH INSTRUCTIONS
                                                        ; WITH THE FOLLOWING CC
173074  100364          PTST1A: BPL     CP1ERR          ; V=0   Z=0     N=1     C=0     PSW=350
173076  002363                  BGE     CP1ERR          ; CHECK ALL THE BRANCH INSTRUCTIONS THAT
173100  003362                  BGT     CP1ERR          ; WILL NOT BRANCH WITH THE CC SET FOR TEST 1A.
173102  002400                  BLT     PTST1B          ; DO THIS BRANCH

                                                        ; PRE TEST 1-B
                                                        ; SET THE CONDITION CODES
173104  000277          PTST1B: SCC                     ; Z=1   N=1     V=1     C=1     PSW=357
173106  001357                  BNE     CP1ERR          ; CHECK ALL THE BRANCH INSTRUCTIONS
173110  100356                  BPL     CP1ERR          ; THAT WILL NOT BRANCH WITH THE
173112  102355                  BVC     CP1ERR          ; CONDITION CODES SET FOR TEST 1B.
173114  103354                  BCC     CP1ERR
173116  003353                  BGT     CP1ERR
173120  101352                  BHI     CP1ERR
173122  002751                  BLT     CP1ERR
173124  002002                  BGE     PTST1C          ; DO BRANCH TO NEXT TEST

                                                        ; DO A BRANCH IN NEAGTIVE DIRECTION.
173126  000270          BRBACK: SEN                     ; SET THE CONDITION CODES
                                                        ; V=0   Z=0     N=1     C=0
173130  000761                  BR      PTST1A          ; DO A BRANCH UNCONDITIONAL
```

```
                                                                ; PRE TEST 1-C
                                                                ; TEST THE JSR INSTRUCTION
173132  022700  173046          PTST1C: CMP     #CP1ERR,R0      ; DID JSR PUT CP1ERR ADDRS IN R0?
173136  001343                          BNE     CP1ERR          ; BRANCH IF ERROR
173140  022706  177522                  CMP     #RWADD,R6       ; DID JSR DECREMENT R6
173144  001340                          BNE     CP1ERR
173146  022737  000340  177522          CMP     #340,@#RWADD    ; DID JSR PUT 340 (R0) INTO RWADD(177522)
173154  001334                          BNE     CP1ERR          ; BRANCH IF ERROR.
173156  005726                          TST     (R6)+           ; MAKE R6= RWADD +2 (177 524)
173160  022706  177524                  CMP     #RWADD+2,R6     ; SO FUTURE JSR WILL WORK.
173164  001330                          BNE     CP1ERR          ; BRANCH IF ERROR.
                                                                ; IF NO ERROR GO TO PRE TEST 1D

                        ;* PRE TEST 1D
                        ;* CHECK ALL SINGLE OPERAND INSTRUCTIONS WITH DESTINATION MODE = 0
                        ;* THIS WILL TEST THE PLA AND SOME FUNCTIONS OF THE INSTRUCTIONS.
                                                                ; REGISTER      NZVC
173166  000277          PTESTD: SCC                             ;               1111
173170  005001                  CLR     R1                      ; 000 000       0100
173172  001041                  BNE     PERRD
173174  101040                  BHI     PERRD
173176  003037                  BGT     PERRD
173200  005101                  COM     R1                      ; 177 777       1001
173202  103035                  BCC     PERRD
173204  003034                  BGT     PERRD
173206  005401                  NEG     R1                      ; 000 001       0001
173210  101032                  BHI     PERRD
173212  002431                  BLT     PERRD
173214  006001                  ROR     R1                      ; 100 000       1001
173216  103027                  BCC     PERRD
173220  100026                  BPL     PERRD
173222  005501                  ADC     R1                      ; 100 001       1000
173224  103424                  BCS     PERRD
173226  000261                  SEC                             ; 100 001       1001
173230  100022                  BPL     PERRD
173232  002021                  BGE     PERRD
173234  006101                  ROL     R1                      ; 000 003       0011
173236  103017                  BCC     PERRD
173240  102016                  BVC     PERRD
173242  006201                  ASR     R1                      ; 000 001       0011
173244  103014                  BCC     PERRD
173246  102013                  BVC     PERRD
173250  005601                  SBC     R1                      ; 000 000       0100
173252  003011                  BGT     PERRD
173254  006301                  ASL     R1                      ; 000 000       0100
173256  002407                  BLT     PERRD
173260  005201                  INC     R1                      ; 000 001       0000
173262  001405                  BEQ     PERRD
173264  005301                  DEC     R1                      ; 000 000       0100
173266  001003                  BNE     PERRD
173270  005701                  TST     R1                      ; 000 000       0100
173272  003001                  BGT     PERRD
```

```
     58 173274  001402                          BEQ     PTESTE          ; RESULT SHOULD BE ZERO
     59
     60
     61                                                  ; ***********************************************
     62 173276  004067  177544          PERRD:  JSR     R0,CP1ERR       ; PRE TEST 1D FAILURE
     63                                                  ; R0=POINTER TO PRE TEST 1D.
     64                                                  ; ERROR REGISTERS
     65                                                  ; R1 = ERROR IN PRE TEST 1-D
     66                                                  ; R2 = 0
     67                                                  ; R3 = 0
     68                                                  ; R4 = 0
     69                                                  ; R5 = 0
     70                                                  ; R6 = RWADD
     71                                                ; ***********************************************
     74
     75                                  .SBTTL  PRE TEST 1E
     78                                  ;* PRE TEST 1E
     79                                  ;* CHECK ALL BYTE SINGLE OPERAND INSTRUCTIONS WITH DESTINATION MODE 0.
     80                                  ;* THIS WILL TEST THE PLA AND SOME FUNCTIONS OF THE INSTRUCTIONS.
     83 173302                          PTESTE:                         ; DIAGNOSTIC LIGHT DISPLAY ='1'
     84
     85                                                                  ; REGISTER      NZVC
     86 173302  112702  000377                  MOVB    #000377,R2      ; 177 777       1000
     87 173306  001441                          BEQ     PERRE
     88 173310  105002                          CLRB    R2              ; 177 400       0100
     89 173312  101037                          BHI     PERRE
     90 173314  105102                          COMB    R2              ; 177 777       1001
     91 173316  103035                          BCC     PERRE
     92 173320  100034                          BPL     PERRE
     93 173322  105602                          SBCB    R2              ; 177 776       1000
     94 173324  103432                          BCS     PERRE
     95 173326  105202                          INCB    R2              ; 177 777       1000
     96 173330  100030                          BPL     PERRE
     97 173332  105402                          NEGB    R2              ; 177 401       0001
     98 173334  101026                          BHI     PERRE
     99 173336  105302                          DECB    R2              ; 177 400       0101
    100 173340  103024                          BCC     PERRE
    101 173342  003023                          BGT     PERRE
    102 173344  000302                          SWAB    R2              ; 000 377       1000
    103 173346  001421                          BEQ     PERRE
    104 173350  106302                          ASLB    R2              ; 000 376       1001
    105 173352  002017                          BGE     PERRE
    106 173354  106002                          RORB    R2              ; 000 377       1010
    107 173356  103415                          BCS     PERRE
    108 173360  003414                          BLE     PERRE
    109 173362  106202                          ASRB    R2              ; 000 377       1001
    110 173364  103012                          BCC     PERRE
    111 173366  102411                          BVS     PERRE
    112 173370  106102                          ROLB    R2              ; 000 377       1001
    113 173372  100007                          BPL     PERRE
    114 173374  001406                          BEQ     PERRE
```

```
    115 173376  105502                      ADCB    R2                      ; 000 000       0101
    116 173400  100404                      BMI     PERRE
    117 173402  001003                      BNE     PERRE
    118 173404  105702                      TSTB    R2                      ; 000 000       0100
    119 173406  103401                      BCS     PERRE
    120 173410  001402                      BEQ     PTESTF                  ; RESULT SHOULD BE 0
    121
    122
    123                                             ; ********************************************
    124 173412  004067  177430      PERRE:  JSR     R0,CP1ERR               ; PRE TEST 1E FAILED
    125                                             ; ERROR REGISTERS
    126                                             ; R0= ADRESS OF ROUTINE IN ERROR.
    127                                             ; R1 = 0
    128                                             ; R2 = PRE TEST 1E ERROR DATA.
    129                                             ; R3 = 0
    130                                             ; R4 = 0
    131                                             ; R5 = 0
    132                                             ; R6 = RWADD
    133                                             ; ********************************************
    138
    139                             .SBTTL  PRE TEST 1F
    142                             ;* PRE TEST 1F
    143                             ;* THIS IS A COMPLETE TEST OF SPECIFIC INSTRUCTIONS USED IN THE SWITCH
    144                             ;* CHECKING ROUTINE. THIS TEST CONCLUDES THE -PRETEST-.
    145                             ;*
    148 173416                      PTESTF:                         ; DIAGNOSTIC LIGHT DISPLAY ="1"
    149 173416  012701  000001              MOV     #BIT0,R1                ; TEST BIT FOR LOCATING BIT13 IN RWADD.
    150 173422  012702  173416              MOV     #PTESTF,R2              ; POINTER TO DATA WHEN BIT MATCH FOUND.
    151 173426  012737  020000  177522      MOV     #BIT13,@#RWADD          ; BIT TESTING FOR.
    152 173434  012703  173502              MOV     #PERR+2,R3              ; POINTER TO CORRECT DATA.
    153
    154 173440  030137  177522      10$:    BIT     R1,@#RWADD              ; TEST IF BIT IN R1 = ONE IN RWADD
    155 173444  001410                      BEQ     20$                     ; BR.IF BIT DOES NOT = ONE IN RWADD
    156
    157 173446  012205                      MOV     (R2)+,R5                ; BIT IN R1 MATCHED BIT IN RWADD
    158 173450  012204                      MOV     (R2)+,R4                ; GET DATA FROM CALCULATER POINTER R2.
    159
    160 173452  022305                      CMP     (R3)+,R5                ; CHECK FOR CORRECT DATA
    161                                                                     ; R5 = 173 046
    162 173454  001011                      BNE     PERR                    ; IF NOT EQUAL GO TO PERR
    163 173456  021304                      CMP     (R3),R4                 ; CHECK THIS ADDRESS
    164 173460  001007                      BNE     PERR                    ; IF NO ERROR GO TO SWITCH TEST
    165                                                                     ; R4 = 012 700
    166
    167 173462  000167  000016              JMP     ENDPRE                  ; END OF PRETEST
    168                                                                     ; TEST JMP LESS THAN 128 LOCATIONS.
    169
    170 173466  022222              20$:    CMP     (R2)+,(R2)+             ; BIT IN R1 DID NOT MATCH BIT IN RWADD
    171 173470  006301                      ASL     R1                      ; BUMP POINTER IN R2 AND SHIFT BIT IN R1
```

```
    172
    173 173472  022701  040000                  CMP     #BIT14,R1               ; SAFETY TEST IF BIT TEST FAILS
    174 173476  001360                          BNE     10$                     ; BR.AS LONG AS BIT IN R1 IS NOT BIT14.
    175
    176                                                 ; ********************************************
    177 173500  004067  177342          PERR:   JSR     R0,CP1ERR               ; ERROR IN PTEST 1F
    178                                                                         ; ERROR REGISTERS
    179                                                                         ; R0= POINTER TO PTEST 1F ERROR.
    180                                                                         ; R1 = TEST BIT .
    181                                                                         ; R2 = CALCULATED DATA POINTER.
    182                                                                         ; R3 = CORRECT DATA POINTER.
    183                                                                         ; R4 = CALCULATED DATA VALUE .
    184                                                                         ; R5 = CALCULATED DATA VALUE.
    185                                                                         ; RWADD = 020 000 (BIT13)
    186                                                 ; ********************************************
    187
    188                                                 ;* GOTO SWITCH CHECKING ROUTINE
    189 173504  012700  000001          ENDPRE: MOV     #BIT0,R0                ; TEST BIT FOR CPU TEST
    190 173510  012701  173550                  MOV     #TABLE, R1              ; ENTRY INTO TABLE
    191
```

```
                        .SBTTL  SWITCH CHECK
                        ;;*****************************************************************************
                        ;;*****************************************************************************
                        ;*
                        ;* THIS ROUTINE WILL CHECK THE SWITCH SETTINGS ON THE TERMINATOR
                        ;* BOARD AND DISPATCH TO THE SUBTEST SELECTED AND TO THE FIRST
                        ;* BOOTSTRAP ENCOUNTERED IN THE TABLE.
                        ;* IF NO LOAD DEVICE IS SELECTED IN THE SWITCH SETTINGS
                        ;* THE CPU WILL HALT. STARTING THE MICRO CODE O.D.T..
                        ;* THIS SUBROUTINE IS CALLED BY THE PRE-TEST,CPU TEST,VT TEST AND MEMORY TEST.
                        ;*
                        ;* REGISTER DEFINITIONS
                        ;* INPUT
                        ;* R0= CONTAINS THE BIT TO BE TESTED IN THE SWITCH SETTINGS
                        ;* R1= CONTAINS THE ENTRY INTO THE TABLE
                        ;* R2= RESULTS OF SOME TEST.
                        ;* R3= RESULTS OF SOME TEST.
                        ;* R4= RESULTS OF SOME TEST.
                        ;* R5= RESULTS OF SOME TEST.
                        ;* R6= RWADD + 2 (177 524)
                        ;*
                        ;* OUTPUT
                        ;* R0= IS THE BIT THAT MATCHED A SWITCH SETTING.
                        ;* R1= NEXT ENTRY INTO TABLE.
                        ;* R2= RESULT OF SOME TEST, NOT USED IN SWITCH CHECK.
                        ;* R3= RESULT OF SOME TEST, NOT USED IN SWITCH CHECK.
                        ;* R4= PAGE CONTROL REGISTER SETTING FOR TEST/BOOT SELECTED.
                        ;* R5= ADDRESS OF ROUTINE TO BE EXECUTED.
                        ;* R6= RWADD + 2 (177 524)
                        ;*
                        ;*                                      FLOPPY  ROM             SLU
                        ;* SWITCH POSITION A1    = CPU1 TEST    -------- ---            ---
                        ;* SWITCH POSITION A2    = MEMORY TEST
                        ;* SWITCH POSITION A3    = SLU BOOT
                        ;* SWITCH POSITION A4    = VT TEST                               DUV-11=1
                        ;* SWITCH POSITION A5    = ROM BOOT                              DLV-11 = 1
                        ;* SWITCH POSITION A6    = FLOPPY BOOT                           OFF. FUTURE USE
                        ;* SWITCH POSITION A7    = RL01 BOOT    RX01= 0
                        ;*                                      RX02= 1
                        ;* SWITCH POSITION A8    = RK-11 BOOT
                        ;* SWITCH POSITION B1    = LOOP ON TEST          DIAG.ROM 2
                        ;* SWITCH POSITION B2                            EPROM
                        ;* SWITCH POSITION B3                            SYS. ROM
                        ;* SWITCH POSITION B4                            B4=0 ROM IS 1K
                        ;*                                               B4=1 ROM IS 2K
                        ;;*****************************************************************************
                        ;;*****************************************************************************
```

```
                                ;
                                ; DIAGNOSTIC LIGHT DISPLAY = "1"
 173514                 SWTEST:
 173514  030037 177524          BIT     R0,@#SWITCH     ; TEST THE SWITCH SETTING
                                ; AND TEST BIT IN R0
 173520  001404          BEQ     20$             ; IF THE SWITCH SETTING DOES NOT
                                ; MATCH GO TO 20$
 173522  012105          MOV     (R1)+,R5        ; TEST BIT MATCHED THE SWITCH SETTING
 173524  012104          MOV     (R1)+,R4        ; PUT THE NEW PC FROM TABLE IN R5
                                ; AND THE NEW PAGE IN R4
 173526  000167 000234          JMP     CPAGE1          ; GO TO ROUTINE SELECTED BY THE
                                ; SWITCH SETTING VIA CHANGE PAGE
                                ;
                                ; TEST BIT DID NOT MATCH.
 173532  022121          20$:    CMP     (R1)+,(R1)+     ; INDEX TO THE NEXT ENTRY IN THE TABLE
 173534  006300          ASL     R0              ; ROTATE THE TEST BIT
 173536  022700 001000          CMP     #BIT9,R0        ; CHECK IF THIS IS LAST BIT TO CHECK
 173542  001364          BNE     SWTEST          ; BRANCH IF THERE ARE MORE BITS TO CHECK
                                ; NO BRANCH IF NO LOAD DEVICE SELLECTED.
                                ; ***********************************************
                                ; ERROR REGISTERS
 173544  004067 177276          JSR     R0,CP1ERR       ; NO LOAD DEVICE SELECTED
                                ; R0 = POINTER TO SWTEST ERROR.
                                ; R1 = ENTRY INTO TABLE
                                ; R2 = RESULT OF SOME TEST/ NOT USED.
                                ; R3 = RESULT OF SOME TEST/ NOT USED.
                                ; R4 = POINTER TO PAGE OF TEST.
                                ; R5 = POINTER TO ADDRESS OF TEST.
                                ; ***********************************************
                                ;
                                ; TABLE FOR SWITCH SETTINGS
 173550  173516          TABLE:  TEST1           ; ADDRESS OF THE CPU TEST 1
 173552  001402                  PAGE2           ; TEST BIT = BIT 0 IN R0
 173554  173000                  MMTEST          ; ADDRESS OF MEMORY TEST
 173556  001402                  PAGE2           ; TEST BIT = BIT 1 IN R0
 173560  173000                  SLBOOT          ; ADDRESS OF THE SLU BOOTSTRAP
 173562  006414                  PAGE14          ; TEST BIT = BIT 2 IN R0
 173564  173302                  VTTEST          ; ADDRESS OF THE VT TEST.
 173566  003406                  PAGE6           ; TEST BIT = BIT 3 IN R0
 173570  173000                  ROBOOT          ; ADDRESS OF THE ROM AREA BOOTSTRAP
 173572  007416                  PAGE16          ; TEST BIT = BIT 4 IN R0
 173574  173000                  RXBOOT          ; ADDRESS OF THE FLOPPY BOOTSTRAP
 173576  005412                  PAGE12          ; TEST BIT = BIT 5 IN R0
 173600  173000                  RLBOOT          ; ADDRESS OF THE RL01 BOOTSTRAP
 173602  004410                  PAGE10          ; TEST BIT = BIT 6 IN R0
 173604  173614                  RKBOOT          ; ADDRESS OF THE RK-11 BOOT.
 173606  000400                  PAGE0           ; TEST BIT = BIT 7 IN R0.
 173610  173050                  LOOP            ; ADDRESS OF THE LOOP TEST.
 173612  000400                  PAGE0           ; TEST BIT = BIT 8 IN R0
```

```
                                .SBTTL  RK-11 BOOT
                                ; ******************************************************************************
                                ; ******************************************************************************
                                ;
                                ; RK -11 BOOTSTRAP
                                ; THIS IS A VERY SIMPLE BOOTSTRAP .
                                ; RK-11 BOOT IS NOT SUPPORTED BY THE SYSTEM BUT IS HERE TO AID IN TESTING.
                                ;
                                ; ******************************************************************************

173614                          RKBOOT:
173614  112737  000012  177524          MOVB    #FIVE,@#LIGHTS          ; INDICATE A LOAD DEVICE SELECTED
173622  012703  000010                  MOV     #10,R3                  ; SET ERROR RETRY COUNTER
173626                          REDORK:
173626  005000                          CLR     R0                      ; R0 = UNIT TO BOOT FROM
173630  012701  177404                  MOV     #177404,R1              ; R1 = CSR OF RKV
173634  005002                          CLR     R2                      ; R2 = WACH DOG TIMER
173636  052700  000010                  BIS     #10,R0                  ; SET THE UNIT REGISTER WITH A TEST BIT
173642  006300                  2$:     ASL     R0                      ; MOVE R0 LEFT UNTIL UNIT BITS
173644  103376                          BCC     2$                      ; IN CORRECT POSITION BITS 13-15
173646  010061  000006                  MOV     R0,6(R1)                ; SET DRIVE ADDRESS REGISTER WITH UNIT BITS AND
                                                                        ; SECTOR=0, SURFACE=0 CYLINDER= 0

173652  032761  000200  177774  4$:     BIT     #BIT7,-4(R1)            ; TEST FOR SELECTED DRIVE READY
173660  001774                          BEQ     4$                      ; LOOP UNTIL IT IS READY.
                                                                ; SET WORD COUNT
173662  012761  177000  000002          MOV     #-256.*2,2(R1)          ; SET WORD COUNT= 177000
173670  012711  000005                  MOV     #5,(R1)                 ; EXECUTE A READ. 5 = GO AND READ COMMAND.
173674                          TDONE:
173674  105711                          TSTB    (R1)                    ; TEST FOR DONE.
173676  100404                          BMI     DONE                    ; BRANCH WHEN DONE.

173700  005202                          INC     R2                      ; INCREMENT WATCHDOG TIMER
173702  005702                          TST     R2                      ; KILL TIME
173704  001373                          BNE     TDONE                   ; IF DONE NOT SET AND THERE IS STILL TOIME
173706  000406                          BR      RKERR                   ; TEST FOR DONE AGAIN.
                                                                        ; IF NO TIME LEFT THEN GO TO ERROR
173710                          DONE:
173710  005711                          TST     (R1)                    ; TEST FOR ERROR BIT 15.
173712  100404                          BMI     RKERR                   ; BRANCH IF SET

173714  112737  000037  177524          MOVB    #BLANK,@#LIGHTS         ; CLEAR THE LIGHTS
173722  005007                          CLR     PC                      ; AND START AT 000000.
173724                          RKERR:
173724  005303                          DEC     R3                      ; DECREMENT THE ERROR RETRY COUNTER
173726  001337                          BNE     REDORK                  ; AFTER 10 RETRYS FLAG ERROR

173730  011003                          MOV     (R0),R3                 ; SET R3 = TO CONTENTS OF CSR
173732  016002  177776                  MOV     -2(R0),R2               ; SET R2 = TO ERROR REGISTER
```

```
     58 173736  032737  003400  177524          BIT #BIT8!BIT9!BIT10,@#SWITCH
     59                                                          ; TEST IF A ROM IS SELECTED
     60 173744  001002                          BNE     20$      ; BRANCH IF ONE IS SELECTED
     61
     62                                                  ; ****************************************************
     63
     64                                                          ; RK ERROR
     65 173746  000000                          HALT             ; R0 = UNIT IN BITS 13- 15.
     66 173750  000721                          BR      RKBOOT   ; R1 = CSR ADDRESS
     67                                                          ; R2 = CONTENTS OF ERROR REGISTER
     68                                                          ; R3 = CONTENTS OF CSR
     69                                                  ; ****************************************************
     70
     71 173752                          20$:
     72 173752  012704  007416                  MOV     #PAGE16,R4       ; SET PAGE OF ROM BOOT IN R4
     73 173756  012705  173000                  MOV     #ROBOOT,R5       ; AND ADDRESS OF ROM BOOT IN R5
     74 173762  000401                          BR      CPAGE1           ; GO CHANGE PAGES.
```

CHANGE PAGE ROUTINE FOR PAGE ONE.

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE ONE.
                                ;;**************************************************************************
                                ;;**************************************************************************
                                ;* CHANGE PAGE LOCATIONS FOR THIS PAGE.
                                ;*
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;*
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;*
                                ;*
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*
                                ;;**************************************************************************
                                ;;**************************************************************************

                                                        ; ZERO FILL REMAINING LOCATIONS WITH HALTS
173764  000000                  .WORD   000000
        173766                  .=173766

173766  010437  177520          CPAGE1: MOV     R4,@#PAGELO             ; LOAD THE NEW PAGE WORD AND
                                                                        ; EXECUTE NEXT INSTRUCTION FROM IT.
173772  010507                          MOV     R5,PC                   ; THIS INSTRUCTION IS THE SAME
                                                                        ; ON ALL PAGES. JAM THE NEW PC.
173774  000000                          .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
173776  000000                          .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
                                                                        ; BY ROM BLASTING PROGRAM.
        000001'                 .END
```

TABLE OF CONTENTS

  1- 9 CPU TESTS
  2- 1 BASIC DEFINITIONS
  3- 1 LOCAL DATA
  4- 1 SPECIAL EQUALITIES FOR BVD11-AA
  5- 1 MEMORY TEST
  8- 1 MEMORY DEPENDENT CPU TESTS.
 11- 1 CPU TEST 1
 13- 1 CPU TEST 2
 15- 1 CPU ERROR HALT TWO.
 15- 24 CHANGE PAGE ROUTINE FOR PAGE THREE.


```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS
                                
173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550
173000                          TEST3   =       173000
000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          . = 173000
```

```
                        .SBTTL  MEMORY TEST
                        ; ******************************************************************************
                        ; ******************************************************************************
                        ;* MEMORY TEST SET UP
                        ;* SET DIAGNOSTIC LIGHT TO 2
                        ;* SET LOW MEMORY LIMIT R0 = 000 000
                        ;* FIND UPPER MEMORY LIMIT FOR THIS SYSTEM = R3
                        ;*
                        ;* MEMORY TEST 1
                        ;* THIS TEST WILL WRITE THE ADDRESS LOCATION INTO ITSELF STARTING AT LOCATION
                        ;* 000 000 AND CONTINUING UNTIL THE UPPER MEMORY LIMIT IS REACHED. CHECKING
                        ;* THE RESULTS WILL BEGIN AT UPPER MEMORY LIMIT AND CHECK EACH LOCATION UNTIL
                        ;* LOCATION 000 000 IS REACHED. A ERROR WILL RESULT IN A "HALT" AT MMERR1.
                        ;*
                        ;* MEMORY TEST 2
                        ;* THE NEXT SUBTEST WILL FIRST FILL ALL OF MEMORY WITH ZERO'S ( 0 ). THEN A WORD
                        ;* OF ONE'S WILL BE WRITTEN AND CHECKED. IF NO ERROR, DO THE NEXT
                        ;* ADDRESS UNTIL ALL ARE CHECKED.A ERROR WILL RESULT IN A "HALT" AT MMERR2.
                        ;*
                        ;* AFTER MEMORY HAS BEEN FILLED WITH ONE'S AND CHECKED THE PROGRAM WILL LOAD A
                        ;* WORD OF ZERO'S INTO A ADDRESS AND CHECK IT. IF THERE ARE NO ERRORS THE NEXT
                        ;* ADDRESS IF LOADED AND CHECKED UNTIL ALL MEMORY LOCATIONS BETWEEN THE LOWER
                        ;* AND UPPER LIMITS ARE CHECKED AND FILLED WITH ZERO'S.
                        ;*
                        ;* MEMORY TEST 3
                        ;* THIS TEST WILL CHECK THE ABILITY OF EACH 1K MEMORY BANK TO DO BYTE ADDRESS
                        ;* TWO BYTES WILL BE WRITTEN INTO THE MEMORY BANK AND THEN CHECKED
                        ;* IF THERE ARE NO ERRORS THEN THE NEXT LOWER 1K MEMORY BANK WILL BE CHECKED.
                        ;* THIS WILL BE DONE TO ALL MEMORY BANKS.
                        ;*
                        ;* THIS TEST WILL THEN JUMP TO THE CPU TESTS FOR INSTRUCTIONS REQUIRING
                        ;* MORE THAN 1 MEMORY ADDRESS.
                        ;*
                        ;* INPUT
                        ;* R0= TEST BIT MATCHED VALUE.
                        ;* R1= NEXT ENTRY INTO TABLE.
                        ;* R2= RESULT OF SOME TEST.
                        ;* R3= RESULT OF SOME TEST.
                        ;* R4= PAGE CONTROL REGISTER SETTING
                        ;* R5= ADDRESS OF MEMORY TEST.
                        ;*
                        ;* OUTPUT
                        ;* R0= TEST BIT SETTING FOR SWITCH CHECKING. (BIT2)
                        ;* R1= ENTRY INTO SWITCH TABLE CHECKING.
                        ;* R2= TEST PATTERN WRITTEN INTO MEMORY.
                        ;* R3= UPPER ADDRESS BOUNDRY OF MEMORY.
                        ;* R4= PAGE CONTROL REGISTER SETTING.
                        ;* R5= ADDRESS OF SWITCH CHECKING ROUTINE.
                        ;*
                        ; ******************************************************************************
                        ; ******************************************************************************
```

```
                                .ENABL LSB
173000                          MMTEST:                 ; ROM PAGE IS PAGE2. 001402 IN PAGE CONTROL REGISTER.

                                                        ; DIAGNOSTIC LIGHT DISPLAY = "2"
                                                                ; MEMORY SET UP.
173000  112737  000015  177524        MOVB    #TWO,@#LIGHTS   ; TURN MEMORY ERROR LIGHT ON.

173006  012706  001000                MOV     #1000,SP        ; SET STACK TO 1000
173012  005000                        CLR     R0              ; R0= IS THE LOW ADDRESS LIMIT.
173014  005003                        CLR     R3              ; R3 WILL CONTAIN THE UPPER MEM LIMIT.

                                                        ; MEMORY LIMIT FOR THE RT-11 SYSTEMS 28K
173016  012702  000006                MOV     #6,R2           ; SETUP ERROR VECTORS FOR MEM TRAP
173022  012712  000340                MOV     #340,(R2)       ; CPU PRIORITY = NO INTERRUPTS ALLOWED
173026  012742  173046                MOV     #5$,-(R2)       ; A NON EXISTANT MEM. WILL CAUSE A TRAP
                                                                ; AND R3 WILL CONTAIN UPPER MEM. LIMIT.

173032  005013                3$:     CLR     (R3)            ; WRITE 0'S INTO LOCATION IN R3.
173034  062703  020000                ADD     #020000,R3      ; GO TO NEXT  4K STACK IF NO TRAP.
173040  022703  160000                CMP     #160000,R3      ; IF 160 000 (28K) THEN THIS IS HIGHEST
173044  001372                        BNE     3$              ; MEMORY ALLOWED ON RT-11.
                                                                ; NO TRAP BUT FALL THROUGH IF R3 = 160 000.

                                                                ; MEMORY TEST 1
173046  005043                5$:     CLR     -(R3)           ; UPPER MEMORY LIMIT VALUE SET IN R3

173050  010001                        MOV     R0,R1           ; LOAD STARTING ADDRESS=0
173052  010121                10$:    MOV     R1,(R1)+        ; AND BEGIN STORING ADDRESS

173054  020103                        CMP     R1,R3           ; IS THIS THE UPPER ADDRESS BOUNDRY?
173056  101775                        BLOS    10$             ; IF NOT CONTINUE WRITING ADDRESS IN ADDRESS
                                                        ; FINISHED WRITING ADDRESS INTO SELF
173060  024101                20$:    CMP     -(R1),R1        ; NOW CHECK THAT EACH ADDRESS
                                                                ; CONTAINS ITS LOCATION
173062  001402                        BEQ     30$             ; IF NO ERROR CONTINUE ON
                                                        ; ********************************************
                                                        ; ERROR IN READING BACK ADDRESS
173064  011104                        MOV     (R1),R4         ; R0= ADDRESS LOW LIMIT
                                                                ; R1= ADDRESS IN ERROR.
                                                                ; R2= NO ERROR INDICATOR
173066  000000                MMERR1: HALT                    ; R3= UPPER MEMORY LIMIT
                                                                ; R4= INCORRECT DATA
                                                                ; R5= NO ERROR INDICATOR
                                                                ; ODT "PROCEED" WILL CONTINUS TEST.
                                                        ; ********************************************
173070  020100                30$:    CMP     R1,R0           ; CHECK IF LOW ADDRESS BOUNDRY
                                                                ; REACHED R0 = 000 000
                                                                ; R1 IS ADDRESS NOW WORKING ON.
173072  001372                        BNE     20$             ; BR.IF LOW LIMIT NOT REACHED.
```

```
                                                        ; TEST 2
173074  005002                    CLR     R2              ; INITIAL TEST PATTERN = 000 000
173076  010221            40$:    MOV     R2,(R1)+        ; LOAD MEMORY WITH BACKGROUND DATA
                                                        ; R2 = DATA
                                                        ; R1 = ADDRESS BEING FILLED.
173100  020103                    CMP     R1,R3           ; FINISHED LOADING MEMORY WITH 0?
                                                        ; R3 IS UPPER ADDRESS BOUNDRY
173102  101775                    BLOS    40$             ; GO BACK IF MORE TO DO
173104  005102            45$:    COM     R2              ; NOW THAT MEMORY IS FILLED
173106  010241            50$:    MOV     R2,-(R1)        ; PUT TEST PATTERN IN MEMORY
173110  020211                    CMP     R2,(R1)         ; CHECK THAT DATA WAS WRITTEN CORRECTLY
173112  001402                    BEQ     60$             ; IF NO ERROR GOTO 60$
173114  011104                    MOV     (R1),R4         ; ERROR IN MEMORY TEST PART 2
                                                        ; R0= LOW MEMORY LIMIT
                                                        ; R1= ADDRESS OF ERROR
                                                        ; R2= EXPECTED DATA PATTERN
173116  000000            MMERR2: HALT                    ; R3= UPPER MEMORY LIMIT
                                                        ; R4= INCORRECT DATA
                                                        ; R5= NO ERROR INDICATOR.
                                                        ; ODT "PROCEED WILL CONTINUE TEST.
173120  020100            60$:    CMP     R1,R0           ; PUT IN NEW BACKGROUND DATA.
                                                        ; HAVE ALL ADDRESS BEEN FILLED?
173122  001371                    BNE     50$             ; HAVE ALL THE MEM TEST BEEN DONE?
173124  010301                    MOV     R3,R1           ; MAKE R1 = UPPER MEM LIMIT.
173126  005702                    TST     R2              ; CHECK IF 1'S AND 0'S HAVE BEEN DONE.
173130  001365                    BNE     45$             ; IF 0'S NOT DONE GO BACK AND DO IT.
                                                        ; GO TO THE CPU TEST 3.
                                                        ; MEMORY TEST 3
173132  012702  125125            MOV     #125125,R2      ; R2= 125 125 TEST PATTERN
                                                        ; LOW  BYTE = 125
                                                        ; HIGH BYTE = 252
173136  110241            70$:    MOVB    R2,-(R1)        ; PUT 125 INTO ADDRS IN R1 (ODD ADDRESS)
173140  000302                    SWAB    R2              ; MAKE R2= 052 652
173142  110241                    MOVB    R2,-(R1)        ; PUT 252 INTO ADDRESS AT R1 (EVEN ADDRS)
173144  020211                    CMP     R2,(R1)         ; CHECK THE WORD JUST WRITTEN TO MEMORY
                                                        ; SHOULD BE 052 652
173146  001402                    BEQ     80$             ; BRANCH IF VALUE COMPARE
173150  011104                    MOV     (R1),R4         ; ERROR IN BYTE MEMORY TEST 3
                                                        ; R0= LOW MEMORY LIMIT.
                                                        ; R1= ADDRESS IN ERROR.
                                                        ; R2=EXPECTED DATA PATTERN
173152  000000            MMERR3: HALT                    ; R3= UPPER MEMORY LIMIT
                                                        ; R4= INCORRECT DATA
                                                        ; ODT "PROCEED" WILL CONTINUS TEST.
173154  000302            80$:    SWAB    R2              ; MAKE THE DATA PATTERN THE ORIGINAL VALUE
                                                        ; AND CLEAR CARRY BIT
173156  162701  002000            SUB     #2000,R1        ; DO NEXT LOWER 1K MEMORY BANK
173162  103365                    BCC     70$             ; BRANCH IF MORE TO DO.
173164  000400                    BR      CP2TST          ; DONE IF NO BRANCH.
                            .DSABL LSB
```

```
                        .SBTTL  MEMORY DEPENDENT CPU TESTS.
                        ; ******************************************************************************
                        ; ******************************************************************************
                        ;* CPU TEST REQUIRING MORE THAN ONE MEMORY LOCATION.
                        ;
                        ;*  TEST WILL BE PERFORMED ON THE FOLLOWING INSTRUCTIONS:
                        ;*  TRAP 377, EMT 377, IOT , BPT , RTT , RTI  INSTRUCTIONS
                        ;*  ILLEGAL INSTRUCTIONS AND BUS ERRORS WILL ALSO BE DONE.
                        ;*
                        ;* TEST SET UP.
                        ;* THIS TEST SETUP WILL SET ALL VECTOR ADDRESS BELOW 1000 TO POINT TO
                        ;* THE TRAP CATCHER HALT LOCATED IN THE ADDRESS FOLLOWING THE VECTOR ADDRESS.
                        ;*
                        ;*      STACK POINTER = 1100
                        ;*
                        ;* "SETVEC"
                        ;* SETUP OF THE VECTOR ADDRESS AND THE PSW FOR THE INSTRUCTION UNDER TEST
                        ;* WILL BE DONE BY THE "SETVEC" ROUTINE.THIS ROUTINE WILL LOAD THE VECTOR
                        ;* ADDRESS OF THE TEST INSTRUCTION WITH A POINTER TO THE ROUTINE "GOTRAP"
                        ;* AND THE VECTOR PSW WITH A VALUE FROM TABLE 1A.
                        ;*
                        ;* "GOTRAP"
                        ;* THE INSTRUCTION UNDER TEST WILL BE EXECUTED AND CONTROL WILL BE PASTED
                        ;* TO THE "GOTRAP" ROUTINE.THIS ROUTINE WILL THEN CHECK THAT THE
                        ;* INSTRUCTION EXECUTED CORRECTLY AND WILL RETURN TO THE MAIN TEST BY
                        ;* EXECUTING A "RTT" OR "RTI" DEPENDING ON THE STATE OF THE CONDITION
                        ;* CODE "CARRY".IF THE INSTRUCTION UNDER TEST TRAPS TO A INCORRECT
                        ;* VECTOR ADDRESS A "HALT" WILL OCCURE AT THE INCORRECT TRAP CATCHER.
                        ;*
                        ;* "RTTEST"
                        ;* THE RTI AND RTT INSTRUCTIONS WILL BE TESTED BY THE ROUTINE "RTTEST"
                        ;* THIS ROUTINE IS CALLED WHEN A RTT OR RTI RETURNS TO THE MAIN TEST
                        ;* FROM "GOTRAP".THIS ROUTINE WILL TEST FOR THE CORRECT EXECUTION OF THE
                        ;* RTT OR RTI INSTRUCTION. THEN THE TEST INSTRUCTION VECTORS WILL
                        ;* BE RESTORED WITH THE TRAP CATCHER HALT AND THE MAIN TEST PROGRAM WILL
                        ;* RESUME.
                        ;*
                        ;*  REGISTER VALUES
                        ;*  R1 = ADDRESS OF TABLE 1A     PSW FOR TRAP INSTRUCTION.
                        ;*  R2 = ADDRESSS OF TABLE 1B    ADDRESS OF 'GOTRAP' ROUTINE.
                        ;*  R3 = ADDRESS OF TABLE 1C     ADDRESS FOLLOWING INSTRUCTION BEING TESTED.
                        ;*  R4 = VECTOR ADDRESS          INSTRUCTION UNDER TEST ADDRESS PLUS "4".
                        ;*  R5 = RESULTS OF READING CPU PSW.
                        ;*  R6 = 1100 INITIAL SETTING.
                        ;*
                        ; ******************************************************************************
                        ; ******************************************************************************
```

```
      1 173166                                CP2TST:                         ; DIAGNOSTIC LIGHT DISPLAY = 1
      2 173166  112737  000016  177524               MOVB    #ONE,@#LIGHTS           ; SET UP LIGHTS.
      3 173174  012701  173460                       MOV     #TABLE1,R1              ; R1 = ADDRESS OF TABLE 1A.
      4                                                                              ;      (PSW FOR INSTRUCTION UNDER TEST).
      5 173200  012102                               MOV     (R1)+,R2                ; R2 = 400 IS UPPER TRAP VECTOR BUFFER.
      6 173202  005042                        10$:   CLR     -(R2)                   ; IN VECTOR ADDRESS BELOW 400 PLACE A
Z     7 173204  010242                               MOV     R2,-(R2)                ; POINTER IN THE VECTOR ADDRESS POINTING
      8 173206  005702                               TST     R2                      ; TO A HALT IN THE ADDRESS FOLLOWING THE
      9 173210  003374                               BGT     10$                     ; VECTOR ADDRESS
     10 173212  012106                               MOV     (R1)+,SP                ; STACK ADDRESS = 1100
     11 173214  012104                               MOV     (R1)+,R4                ; R1 = ADDRESS OF TABL 1A.
     12                                                                              ;      (PSW FOR INSTRUCTION UNDER TEST).
     13 173216  012102                               MOV     (R1)+,R2                ; R2 = ADDRESS OF TABL 1B.
     14                                                                              ;      (ADDRESS OF "GOTRAP")
     15 173220  012103                               MOV     (R1)+,R3                ; R3 = ADDRESS  OF TABL 1C (ADDRESS
     16                                                                              ;       FOLLOWING INSTRUCTION UNDER TESTED)
     17                                                                              ; R4 = POINTER TO VECTOR ADDRESS
     18                                                                              ;      BEING TESTED PLUS "4".
     20 173222  004767  000170                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.34)=GOTRAP.
     21 173226  104777                               TRAP 377                        ; SETVEC=PSW FOR"TRAP"= 201 IN LOC.36
     22 173230  004767  000132               RETUR1: JSR     PC,RTTEST               ; GOTRAP CALLED BY"TRAP"
     23                                                                              ; GOTRAP WILL CHECK RESULTS OF"TRAP".
     24                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
     26 173234  004767  000156                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.30)=GOTRAP.
     27 173240  104377                               EMT 377                         ; SETVEC=PSW FOR"EMT"= 201 IN LOC.32
     28 173242  004767  000120               RETUR2: JSR     PC,RTTEST               ; GOTRAP CALLED BY"EMT".
     29                                                                              ; GOTRAP WILL CHECK RESULTS OF "EMT".
     30                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
     32                                                                              ; MOVE POINTER OVER POWER FAIL
     33 173246  024444                               CMP     -(R4),-(R4)             ; POWER FAIL AT 24 - 26.
     34
     35 173250  004767  000142                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.20)=GOTRAP.
     36 173254  000004                               IOT                             ; SETVEC=PSW FOR"IOT"= 210 IN LOC.22
     37 173256  004767  000104               RETUR3: JSR     PC,RTTEST               ; GOTRAP CALLED BY"IOT"
     38                                                                              ; GOTRAP WILL CHECK RESULTS OF"IOT".
     39                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
     41 173262  004767  000130                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.14)=GOTRAP.
     42 173266  000003                               BPT                             ; SETVEC=PSW FOR"BPT"= 204 IN LOC.16
     43 173270  004767  000072               RETUR4: JSR     PC,RTTEST               ; GOTRAP CALLED BY"BPT"
     44                                                                              ; GOTRAP WILL CHECK RESULTS OF"BPT".
     45                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
     47 173274  004767  000116                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.10)=GOTRAP.
     48 173300  107777                               107777                          ; SETVEC=PSW FOR"RESERVED INST"= 202 IN LOC.12
     49 173302  004767  000060               RETUR5: JSR     PC,RTTEST               ; GOTRAP CALLED BY" RESERVED INST."
     50                                                                              ; GOTRAP WILL CHECK RESULTS OF"RESERVED INST.".
     51                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
     53 173306  004767  000104                       JSR     PC,SETVEC               ; SETVEC= SETUP VECTOR ADDRS(LOC.4)=GOTRAP.
     54 173312  000101                               JMP     R1                      ; SETVEC=PSW FOR"ILL INST."= 201 IN LOC.6
     55 173314  004767  000046               RETUR6: JSR     PC,RTTEST               ; GOTRAP CALLED BY"ILL. INST."
     56                                                                              ; GOTRAP WILL CHECK RESULTS OF"ILL. INST.".
     57                                                                              ; RTTEST WILL CHECK RESULTS OF RTI OR RTT.
```

```
                                                        ; END OF TEST GO TO MEMORY TEST.
173320  000444                    BR      MMEND         ; SET UP TO CHANGE PAGES.
173322                    GOTRAP:                               ; CHECK RESULT OF TRAPS
173322  106705                    MFPS    R5                    ; DID INSTRUCTION GET CORRECT PSW FROM
                                                              ; VECTOR DATA.STORE PSW IN R5
173324  020627  001074            CMP     SP,#1074              ; DID INSTRUCTION DECREMENT SP
173330  001014                    BNE     ERCP2A                ; TRAPS = (1100 - 4 = 1074)
173332  022316                    CMP     (R3)+,(SP)            ; DID INSTRUCTION PUT CORRECT PC ON STACK
173334  001012                    BNE     ERCP2A                ; R3 = TABLE 1C CORRECT PC VALUE
173336  022766  000340  000002    CMP     #340,2(SP)            ; DID INSTRUCTION PUT CORRECT PSW ON STACK
173344  001006                    BNE     ERCP2A                ; 340 WAS PSW WHEN TRAP OCCURED.
                                                              ; CHECK THE PSW STORED IN R5
173346  120521                    CMPB    R5,(R1)+              ; R5 = PSW
173350  001004                    BNE     ERCP2A                ; R1 POINTS TO CORRECT PSW
                                                              ; STORED IN TABLE 1A.
173352  005705                    TST     R5                    ; IF CARRY SET DO A RTI
173354  103401                    BCS     10$                   ; IF CARRY NOT SET DO A RTT
173356  000006                    RTT                           ; CARRY = 01
173360  000002            10$:    RTI
                                                        ; ************************************************
173362  004067  000344    ERCP2A: JSR     R0,CP2ERR             ; ERROR IN GOTRAP
                                                              ; R0 = LOCATION OF TEST THAT FAILED
                                                              ; R1 = POINTER TO CORRECT PSW.
                                                              ;       TRAP = TABLE 1A + 0
                                                              ;       EMT  = TABLE 1A + 1
                                                              ;       IOT  = TABLE 1A + 2
                                                              ;       BPT  = TABLE 1A + 3
                                                              ;       RES INST =  TABLE 1A + 4
                                                              ;       ILL INST =  TABLE 1A + 5
                                                              ; R2 = POINTER TO GOTRAP.
                                                              ; R3 = POINTER TO CORRECT RETURN ADDRESS.
                                                              ;       TRAP = TABLE 1C + 0
                                                              ;       EMT  = TABLE 1C + 2
                                                              ;       IOT  = TABLE 1C + 4
                                                              ;       BPT  = TABLE 1C + 6
                                                              ;       RES INST =  TABLE 1C + 10
                                                              ;       ILL INST =  TABLE 1C + 12
                                                              ; R4 = VECTOR ADDRESS UNDER TEST PLUS 4.
                                                              ;       TRAP    =       40
                                                              ;       EMT     =       34
                                                              ;       IOT     =       24
                                                              ;       BPT     =       20
                                                              ;       RES INST=       14
                                                              ;       ILL INST=       10
                                                              ; R5 = READ CPU PSW VALUE.
                                                              ;       TRAP    =       201
                                                              ;       EMT     =       201
                                                              ;       IOT     =       210
                                                              ;       BPT     =       204
                                                              ;       RES INST=       202
```

```
                                                                        ;       ILL INST=       201
                                                                        ; R6 =  RWADD ADDRESS
                                                                        ;
173366                                  RTTEST:                         ; TEST RESULT OR RTT OR RTI
173366  106705                                  MFPS    R5              ; CHECK THAT INSTRUCTION CORRECTLY RESTORED
173370  120527  000340                          CMPB    R5,#340         ; THE PSW FROM STACK
173374  001006                                  BNE     ERCP2B
                                                                        ;
173376  020627  001076                          CMP     SP,#1076                ; DID INSTRUCTION INCREMENT SP CORRECTLY?
173402  001003                                  BNE     ERCP2B
                                                                        ;
173404  005044                                  CLR     -(R4)           ; CLEAR VECTOR ADDRESS AND SET UP
Z 173406  010444                                MOV     R4,-(R4)        ; HALT FOLLOWING VECTOR ADDRESS
173410  000207                                  RTS     PC              ; RETURN
                                                                        ;
173412  004067  000314          ERCP2B: JSR     R0,CP2ERR               ; ERROR IN RTTEST ROUTINE
                                                                        ; R0 = LOCATION OF TEST THAT FAILED
                                                                        ; R1 = POINTER TO CORRECT PSW.
                                                                        ; R2 = POINTER TO GOTRAP.
                                                                        ; R3 = POINTER TO CORRECT RETURN ADDRESS.
                                                                        ; R4 = VECTOR ADDRESS UNDER TEST.
                                                                        ; R5 = READ CPU PSW VALUE.
                                                                        ; R6 =  RWADD ADDRESS
                                                                        ;
173416                                  SETVEC:                         ; SETUP VECTOR ADDRESS AND
                                                                        ; PSW FOR INSTRUCTION TO BE TESTED.
173416  111164  177776                          MOVB    (R1),-2(R4)     ; R1 = ADDRESS OF TABL1A=
173422  011264  177774                          MOV     (R2),-4(R4)     ; R2 = ADDRESS OF TABL1B=GOTRAP
                                                                        ; R4 = VECTOR ADDRESS TO BE LOADED
                                                                        ;       WITH VECTOR ADDRESS AND NEW PSW
173426  000257                                  CCC                     ; CLEAR THE CONDITION CODES.
173430  000207                                  RTS     PC
                                                                        ;
                                                                        ;
                                                                        ; RETURN TO SWITCH CHECK ROUTINE.
173432                                  MMEND:
173432  012706  177524                          MOV     #RWADD+2,SP             ; RESTORE STACK.
173436  012700  000004                          MOV     #BIT2,R0                ; SETUP FOR SWITCH CHECKING
173442  012701  173560                          MOV     #TABLE+^O10,R1          ; R0= CHECK BIT FOR SLU BOOT
                                                                                ; R1= INDEX INTO TABLE
173446  012704  000400                          MOV     #PAGE0,R4               ; GO BACK TO PAGE 0
173452  012705  173514                          MOV     #SWTEST,R5              ; ADDRESS TO GO TO.
173456  000543                                  BR      CPAGE3                  ; GO DO THE CHANGE PAGE CALL
```

```
 1 173460                       TABLE1:
 2 173460  000400                        400                    ; UPPER LIMIT FOR TRAP VECTOR BUFFER AREA
 3 173462  001100                       1100                    ; STACK POINTER = 1100
 4 173464  000040                         40                    ; POINTER FOR VECTOR BEING TESTED
 5 173466  173500                     TABL1B                    ; ADDRESS OF TABL1B
 6 173470  173502                     TABL1C                    ; ADDRESS OF TABL1C
 7
 8 173472     201               TABL1A: .BYTE   201             ; PSW FOR "TRAP"
 9 173473     201                       .BYTE   201             ; PSW FOR "EMT"
10
11 173474     210                       .BYTE   210             ; PSW FOR "IOT"
12 173475     204                       .BYTE   204             ;          BPT
13 173476     202                       .BYTE   202             ;          ILLEGAL INSTRUCTION
14 173477     201                       .BYTE   201             ;          TIMEOUT
15
16                             .EVEN
17 173500  173322              TABL1B: GOTRAP
18
19 173502  173230              TABL1C: RETUR1                  ; CHECK PC FOR TRAP
20 173504  173242                      RETUR2                  ;              EMT
21 173506  173256                      RETUR3                  ;              IOT
22 173510  173270                      RETUR4                  ;              BPT
23 173512  173302                      RETUR5                  ;              ILLEGAL INSTRUCTION
24 173514  173314                      RETUR6                  ;              TIMEOUT
```

```
                        .SBTTL  CPU TEST 1
                        ;* TEST 1
                        ;* THIS TEST WILL CHECK THE "MOVE" AND COMPARE "WORD" INSTRUCTIONS
                        ;* USING THE FOLLOWING SOURCE MODES (SM) AND DESTINATION MODE (DM).
                        ;* MOVE - WORD - DESTINATION MODE - EXECUTION - CALCULATION -
                        ;* THE MOVE DESTINATION CALCULATION AND EXECUTION MODE WILL BE DM = 0
                        ;* MOVE - COMPARE - SM - - CALCULATION
                        ;* THE SOURCE MODES FOR THE MOVE AND COMPARE WILL USE THE SAME MICRO-CODE
                        ;* ROUTINES FOR THE CALCULATION.
                        ;* COMPARE - DM - EXECUTION -
                        ;* THE DESTINATION MODE WORD "COMPARE" INSTRUCTION WILL USE THE SAME MICRO
                        ;* CODE CALCULATION ROUTINE BUT THE EXECUTION WILL BE ALL MODE 0.
                        ;*
                        ;* DM=0
                        ;*      THE DESTINATION MODES WILL BE MODE ZERO (0) FOR ALL OF TEST 1.
                        ;*      THE CALCULATION AND THE EXECUTION FOR A WORD IS A SEPERATE
                        ;*      MICRO CODE ROUTINE.
                        ;*
                        ;* TEST THE PC ADDRESSING MODES.
                        ;* THE SOURCE MODE CALCULATION ROUTINES ARE COMMON TO ALL WORD INSTRUCTIONS.
                        ;*
                        ;* SM=27 (#N)
                        ;*      IMMEDIATE MODE OPERAND IS LOCATED IN THE LOCATION IMMEDIATELY
                        ;*      FOLLOWING THE INSTRUCTION.
                        ;*
                        ;* SM=37 (@#A)
                        ;*      ABSOLUTE ADDRESSING
                        ;*      THE LOCATION FOLLOWING THE INSTRUCTION IS THE ADDRESS OF
                        ;*      THE OPERAND.
                        ;*
                        ;* SM=67 (A)
                        ;*      RELATIVE ADDRESS
                        ;*      THE LOCATION FOLLOWING THE INSTRUCTION IS USED AS A INDEX
                        ;*      TO LOCATE THE SELECTED LOCATION
                        ;*
                        ;* SM=77 OPR @X(PC)
                        ;*      RELATIVE DEFERED ADDRESSING
                        ;*      POINTS TO THE ADDRESS OF THE ADDRESS
                        ;*
                        ;* REGISTER  VALUES
                        ;* INPUT
                        ;* R0= SWITCH SETTING FOR CPU TEST.
                        ;* R1= NEXT ENTRY INTO SWITCH CHECKING TABLE.
                        ;* R2= UNKNOWN TEST RESULT.
                        ;* R3= UNKNOWN TEST RESULT.
                        ;* R4= PAGE SETTING FOR CPU TESTS.
                        ;* R5= ADDRESS OF CPU TEST.
                        ;* R6= RWADD ADDRESS PLUS 2.
```

```
 1 173516                          CP1TST:                 ; DIAGNOSTIC LIGHT DISPLAY = 1.
 2 173516                          TEST1:                  ; SET TO A 1 IN THE PRE-TEST.
 3 173516  012701  173720                  MOV     #CHOSE,R1               ; R1 IS ADDRESS OF CHOSE SM = 27
 4 173522  022701  173720                  CMP     #CHOSE,R1               ; CHECK DATA IN R1       SM = 27
 5 173526  001014                          BNE     ERR1                    ;
 6
 7 173530  013702  173720                  MOV     @#CHOSE,R2              ; R2=DATA IN CHOSE       SM = 37
 8 173534  016703  000160                  MOV     CHOSE,R3                ; R3=DATA IN CHOSE       SM = 67
 9 173540  021102                          CMP     (R1),R2                 ; CHECK DATA IN R2       SM = 1
10 173542  001006                          BNE     ERR1                    ; R1= ADDRESS OF CHOSE
11 173544  021103                          CMP     (R1),R3                 ; CHECK DATA IN R3       SM = 1
12 173546  001004                          BNE     ERR1
13
14 173550  017704  000144                  MOV     @CHOSE,R4               ; R4=DATA IN TABLE 2     SM = 77
15 173554  021204                          CMP     (R2),R4                 ; CHECK DATA IN TABLE 2  SM = 1
16 173556  001402                          BEQ     TEST2                   ; R2= ADDRESS OF TABLE 2
17
18                                                                  ; ***********************************************
19 173560  004067  000146          ERR1:   JSR     R0,CP2ERR               ; ERROR IN TEST 1
20                                                                  ; R0 = ERROR POINTER.
21                                                                  ; R1 = ADDRESS OF CHOSE
22                                                                  ; R2 = ADDRESS OF TABLE 2
23                                                                  ; R3 = ADDRESS OF TABLE 2
24                                                                  ; R4 = DATA IN TABLE 2
25                                                                  ; R5 = NOT USED IN THIS TEST.
26                                                                  ; ***********************************************
```

```
                        .SBTTL  CPU TEST 2
                        ;* TEST 2
                        ;* THIS TEST WILL CHECK THE "MOVE" AND "COMPARE" "WORD" INSTRUCTIONS USING
                        ;* THE FOLLOWING REGISTER ADDRESSING MODES. THE MICRO CODE ROUTINES FOR
                        ;* THE SOURCE MODES (SM) ARE COMMONLY USED BY ALL INSTRUCTIONS.
                        ;* COMPARE - MOVE - SM - CALCULATION -
                        ;* THE MICRO CODE ROUTINES FOR THE SM CALCULATION WILL BE THE SAME FOR THE
                        ;* MOVE AND COMPARE INSTRUCTIONS.
                        ;* COMPARE - DM - CALCULATION -
                        ;* THE MICRO CODE FOR THE COMPARE DM CALCULATION WILL BE THE SAME AS THE SM.
                        ;* MOVE - DN - CALCULATION/EXECUTION
                        ;* THE MICRO CODE ROUTINES WILL BE THE SAME AS IN TEST 5 MOVE DM = 0.
                        ;* DM=0 R        DESTINATION MODE WILL BE ZERO (0) FOR ALL OF TEST 2.
                        ;* SM=0 R         THE SPECIFIED REGISTER CONTAINS THE OPERAND.
                        ;* SM=1 (R)
                        ;*      REGISTER DEFERED -
                        ;*       THE SPECIFIED REGISTER CONTAINS THE ADDRESS OF THE OPERAND.
                        ;* SM=2 (R)+
                        ;*      AUTOINCREMENT -
                        ;*       THE SPECIFIED REGISTER CONTAINS THE ADDRESS OF THE
                        ;*       OPERAND,THE ADDRESS IS THEN BUMPED.
                        ;* SM=3 @(R)+
                        ;*      AUTOINCREMENT DEFERED -
                        ;*       THE REGISTER IS A POINTER TO A WORD CONTAINING THE
                        ;*       ADDRESS OF THE OPERAND. THE POINTER IS THEN BUMPED.
                        ;* SM=4 -(R)
                        ;*      AUTODECREMENT -
                        ;*       THE REGISTER IS DECREMENTED AND THEN USED AS A POINTER.
                        ;* SM=5 @-(R)
                        ;*      AUTODECREMENT DEFERED
                        ;*       REGISTER IS DECREMENTED AND USED TO POINT TO A WORD
                        ;*       CONTAINING THE ADDRESS OF THE OPERAND.
                        ;* SM=6 X(R)
                        ;*      INDEX
                        ;*       THE INDEX VALUE IS ADDED TO THE ADDRESS STORED IN THE REGISTER
                        ;*       THIS VALUE POINTS TO THE OPERAND.
                        ;* SM=7 @X(R)
                        ;*      INDEX DEFERED
                        ;*       THE INDEX VALUE IS ADDED TO THE REGISTER. THIS VALUE
                        ;*       IS USED AS A POINTER TO A WORD CONTAINING THE ADDRESS OF THE OPERAND.
```

```
                                ;* REGISTER  VALUES
                                ;* INPUT
                                ;* R0= ADDRESS OF CHOSE.
                                ;* R1= ADDRESS OF TABLE 2.
                                ;* R2= ADDRESS OF TABLE 2.
                                ;* R3= DATA AT TABLE 2. 125 252
                                ;* R4= PAGE SETTING FOR CPU TESTS.
                                ;* R5= ADDRESS OF CPU TEST.
                                ;* R6= 177 544
                                ;* OUTPUT
                                ;* R0= ADDRESS OF CHOSE2.
                                ;* R1= 125 252
                                ;* R2= ADDRESS OF RWADD.
                                ;* R3= 052 525
                                ;* R4= 125 252
                                ;* R5= 052525
                                ;* R6= 177 544
                                ;*
173564                          TEST2:
173564  012700  173720                MOV     #CHOSE,R0               ; R1= ADDRESS OF TABLE 2        SM=27
173570  012001                        MOV     (R0)+,R1                ; R2= ADDRESS OF TABLE 3        SM=2
173572  012002                        MOV     (R0)+,R2                ; R0= ADDRESS OF TABLE 2        SM=2
173574  022700  173724                CMP     #TABLE2,R0              ; CHECK THAT R0 IS CORRECT      SM=27
173600  001045                        BNE     ERR2                    ; R0 = ADDRESS OF TABLE 2
173602  011203                        MOV     (R2),R3                 ; R3= DATA IN TABLE 3           SM=1
173604  022703  052525                CMP     #052525,R3              ; R3= 052 525                   SM=27
173610  001041                        BNE     ERR2                    ; R2= ADDRESS OF TABLE 3
173612  012204                        MOV     (R2)+,R4                ; R4= DATA IN TABLE 3           SM=3
173614  020403                        CMP     R4,R3                   ; R4= 052 525                   SM=0
173616  001036                        BNE     ERR2                    ; R2= ADDRESS OF TABLE 3A
173620  012204                        MOV     (R2)+,R4                ; R4= DATA IN TABLE 3A          SM=3
173622  022704  125252                CMP     #125252,R4              ; R4= 125 252                   SM=27
173626  001032                        BNE     ERR2                    ; R2= ADDRESS OF TABLE 3B
173630  014205                        MOV     -(R2),R5                ; R5= DATA IN TABLE 3B          SM=4
173632  020405                        CMP     R4,R5                   ; R5= 125 252                   SM=0
173634  001027                        BNE     ERR2                    ; R2= ADDRESS OF TABLE 3A
173636  014205                        MOV     -(R2),R5                ; R5= DATA IN TABLE 3           SM=4
173640  020503                        CMP     R5,R3                   ; R5= 052 525                   SM=0
173642  001024                        BNE     ERR2                    ; R2= ADDRESS OF TABLE 3
173644  015001                        MOV     @-(R0),R1               ; R0=ADDRESS OF CHOSE 1         SM=5
173646  020103                        CMP     R1,R3                   ; R1= DATA IN TABLE 3           SM=0
173650  001021                        BNE     ERR2                    ; R1= 052 525
173652  015001                        MOV     @-(R0),R1               ; R0= ADDRESS OF CHOSE          SM=5
173654  020401                        CMP     R4,R1                   ; R1= DATA IN TABLE 2           SM=0
173656  001016                        BNE     ERR2                    ; R1= 125 252
```

```
     58 173660  013002                          MOV     @(R0)+,R2               ; R2= DATA IN TABLE 2           SM=3
     59 173662  020204                          CMP     R2,R4                   ; R2= 125 252                   SM= 0
     60 173664  001013                          BNE     ERR2                    ; R0= ADDRESS OF CHOSE 1
     61
     62 173666  013002                          MOV     @(R0)+,R2               ; R2= DATA IN TABLE 3           SM=3
     63 173670  020302                          CMP     R3,R2                   ; R2= 052 525                   SM=1
     64 173672  001010                          BNE     ERR2                    ; R0= ADDRESS OF TABLE 2
     65
     66 173674  016002  000004                  MOV     4(R0),R2                ; R2= DATA IN TABLE 2           SM=6
     67 173700  020102                          CMP     R1,R2                   ; R2= 125 252                   SM=0
     68 173702  001004                          BNE     ERR2                    ; R0= ADDRESS OF TABLE 2
     69
     70 173704  017002  177776                  MOV     @-2(R0),R2              ; R2= DATA IN TABLE 3           SM=7
     71 173710  020302                          CMP     R3,R2                   ; R2= 052 525 R3= 052 525       SM=27
     72 173712  001415                          BEQ     NEXT                    ; BRANCH TO NEXT TEST IF NO ERROR
     73
     74                                                                  ; ***********************************************
     75 173714  004067  000012          ERR2:   JSR     R0,CP2ERR               ; ERROR IN TEST 2
     76                                                                  ; GO TO CPU TEST 2 ERROR.
     77                                                                  ; R0 WILL CONTAIN A POINTER TO HERE.
     78                                                                  ; ***********************************************
     79
     80
     81 173720  173724                  CHOSE:  TABLE2                  ;TABLE2
     82 173722  173726                  CHOSE1: TABLE3
     83
     84 173724  125252                  TABLE2: 125252
     85
     86 173726  052525                  TABLE3: 052525
     87 173730  125252                  TABL3A: 125252
```

```
                                .SBTTL  CPU ERROR HALT TWO.
                                ;* ERROR HALT FOR THIS PAGE.
                                ;* R0 = ADDRESS OF ROUTINE IN ERROR.
                                ;* R1 TO R5 WILL CONTAIN THE ERROR POINTERS FOR THE ROUTINE IN ERROR.

173732  000000                  CP2ERR: HALT                            ; CPU TEST ERROR ON THIS PAGE
                                                                        ; R0 CONTAINS POINTER TO ERROR LOCATION.
173734  012705  173000                  MOV     #START,R5               ; ODT PROCEED WILL RESTART ROM CODE
173740  012704  000400                  MOV     #PAGE0,P4               ; AT START ON PAGE 0
173744  000410                        BR      CPAGE3
                                ;*
                                ;* INSTRUCTIONS TO SET UP R4 AND R5 TO POINT TO THE NEXT TEST ON A NEW PAGE.
                                ;*
                                ;*
173746  012705  173000          NEXT:   MOV     #TEST3,R5               ; GO TO TEST 3 . CONTINUS CPU TEST
173752  012704  002404                  MOV     #PAGE4,R4               ; SET UP CORRECT PAGE
173756  000403                        BR      CPAGE3                  ; GO CHANGE PAGE.


                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE THREE.
                                ;* CHANGE PAGE ROUTINE FOR THIS PAGE.
                                ;*
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;*
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;*
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*
                                ;*********************************************************************************
                                ;*********************************************************************************


                                ; FILL REMAINING ROM LOCATIONS WITH HALTS.

173760  000000                  .WORD   000000
173762  000000                  .WORD   000000
173764  000000                  .WORD   000000
        173766                  .=173766

173766  010437  177520          CPAGE3: MOV     R4,@#PAGELO             ; LOAD THE NEW PAGE WORD AND
                                                                        ; EXECUTE NEXT INSTRUCTION FROM IT.
173772  010507                        MOV     R5,PC                   ; THIS INSTRUCTION IS THE SAME
                                                                        ; ON ALL PAGES. JAM THE NEW PC.
173774  000000                        .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
173776  000000                        .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
                                                                        ; BY ROM BLASTING PROGRAM.
        000001'                 .END
```

TABLE OF CONTENTS

  1- 9 CPU TESTS
  2- 1 BASIC DEFINITIONS
  3- 1 LOCAL DATA
  4- 1 SPECIAL EQUALITIES FOR BVD11-AA
  5- 1 CPU TEST 3
  7- 1 CPU TEST 4
  9- 1 CPU TEST 5
 11- 1 CPU TEST 6
 12- 1 CPU TEST 7
 14- 15 CPU ERROR HALT THREE.
 14- 30 CHANGE PAGE ROUTINE FOR PAGE FIVE.


```
                                ;*BASIC "CPU" TRAP VECTOR ADDRESSES
000004                          ERRVEC= 4               ;;TIME OUT AND OTHER ERRORS
000010                          RESVEC= 10              ;;RESERVED AND ILLEGAL INSTRUCTIONS
000014                          TBITVEC=14              ;;"T" BIT
000014                          TRTVEC= 14              ;;TRACE TRAP
000014                          BPTVEC= 14              ;;BREAKPOINT TRAP (BPT)
000020                          IOTVEC= 20              ;;INPUT/OUTPUT TRAP (IOT) **SCOPE**
000024                          PWRVEC= 24              ;;POWER FAIL
000030                          EMTVEC= 30              ;;EMULATOR TRAP (EMT) **ERROR**
000034                          TRAPVEC=34              ;;"TRAP" TRAP
000060                          TKVEC=  60              ;;TTY KEYBOARD VECTOR
000064                          TPVEC=  64              ;;TTY PRINTER VECTOR
000240                          PIRQVEC=240             ;;PROGRAM INTERRUPT REQUEST VECTOR



                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS

173000                          TEST8   =       173000
173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550

000000                          .ASECT                  ; SET ABSOLUTE ADDRESSING MODE.
173000                          . = 173000
```

DIGITAL EQUIPMENT CORPORATION MAYNARD, MASSACHUSETTS	TITLE: BDV11-AA DIAGNOSTIC/BOOT ROM LISTING	SHEET 35 OF 118	SIZE: K	CODE: CS	NUMBER: BDV11-A-2	REV:

```
                        .SBTTL  CPU TEST 3
                        ;;*****************************************************************
                        ;;*****************************************************************
                        ;* TEST 3
                        ;* TEST THE MOVE AND COMPARE WITH A MIXTURE OF SOURCE MODES AND
                        ;* DESTINATION MODES 1 TO 7.
                        ;*
                        ;* TO USE OTHER DESTINATION MODE 0 THE READ/WRITE REGISTER MUST
                        ;* BE USED.
                        ;*
                        ;*
                        ;* SM=?
                        ;*      DESTINATION MODE IS NOT RESTRICTED IN THIS TEST, IT CAN BE ANYTHING.
                        ;*
                        ;* DM=0 R
                        ;*      REGISTER -
                        ;*       THE SPECIFIED REGISTER CONTAINS THE OPERAND.
                        ;*
                        ;* DM=1 (R)
                        ;*      REGISTER DEFERED -
                        ;*       THE SPECIFIED REGISTER CONTAINS THE ADDRESS OF THE OPERAND.
                        ;*
                        ;* DM=2 (R)+
                        ;*      AUTOINCREMENT -
                        ;*       THE SPECIFIED REGISTER CONTAINS THE ADDRESS OF THE
                        ;*       OPERAND,THE ADDRESS IS THEN BUMPED.
                        ;*
                        ;* DM=3 @(R)+
                        ;*      AUTOINCREMENT DEFERED -
                        ;*       THE REGISTER IS A POINTER TO A WORD CONTAINING THE
                        ;*       ADDRESS OF THE OPERAND. THE POINTER IS THEN BUMPED.
                        ;*
                        ;* DM=4 -(R)
                        ;*      AUTODECREMENT -
                        ;*       THE REGISTER IS DECREMENTED AND THEN USED AS A POINTER.
                        ;*
                        ;* DM=5 @-(R)
                        ;*      AUTODECREMENT DEFERED
                        ;*       REGISTER IS DECREMENTED AND USED TO POINT TO A WORD
                        ;*       CONTAINING THE ADDRESS OF THE OPERAND.
                        ;*
                        ;*
                        ;* DM=6 X(R)
                        ;*      INDEX
                        ;*       THE INDEX VALUE IS ADDED TO THE ADDRESS STORED IN THE REGISTER
                        ;*       THIS VALUE POINTS TO THE OPERAND.
                        ;*
                        ;* DM=7 @X(R)
                        ;*      INDEX DEFERED
                        ;*       THE INDEX VALUE IS ADDED TO THE REGISTER. THIS VALUE
                        ;*       IS USED AS A POINTER TO A WORD CONTAINING THE ADDRESS
                        ;*       OF THE OPERAND.
```



```
                        ;*
```

```
                        ;* REGISTER  VALUES
                        ;* INPUT
                        ;* R0= ADDRESS OF TBL1.
                        ;* R1= 052525
                        ;* R2= ADDRESS OF CHOS.
                        ;* R3= 125252
                        ;* R4= ADDRESS OF TABL 3
                        ;* R5= 052525
                        ;* OUTPUT
                        ;* R0= RWADD MINUS 4
                        ;* R1= RWADD ADDRESS
                        ;* R2= ADDRESS OF CHOS
                        ;* R3= 125 252
                        ;* R4= ADDRESS OF TABL 3
                        ;* R5= 052 525
173000                  TEST3:                  ; ROM PAGE IS PAGE4, 002404 IN PAGE CONTROL REGISTER.
                                                ; DIAGNOSTIC LIGHT DISPLAY = 1.

173000  012700  173564          MOV     #CHOS,R0            ; R1= ADDRESS OF TABL 1 DM=0
173004  012001                  MOV     (R0)+,R1            ; R2= ADDRESS OF TABL 2
173006  012002                  MOV     (R0)+,R2            ; R3= ADDRESS OF TABL 3
173010  012003                  MOV     (R0)+,R3            ; R4= ADDRESS OF RWADD(177542)
173012  011304                  MOV     (R3),R4             ; R5= ADDRESS OF TABL 3
173014  010305                  MOV     R3,R5               ; R0= ADDRESS OF TABL 1

                                                ; TEST INDEX MODE ADDRESSING
173016  011172  000010          MOV     (R1),@10(R2)        ; RWADD=DATA IN TABL 1          DM=7
173022  022170  000020          CMP     (R1)+,@20(R0)       ; RWADD = 125 252               DM=7
173026  001040                  BNE     ERR3                ; R1= ADDRS OF TABL 1A

173030  011166  177776          MOV     (R1),-2(R6)         ; RWADD= DATA IN TABL 1A        DM=6
173034  022166  177776          CMP     (R1)+,-2(R6)        ; RWADD= 052 525                DM=6
173040  001033                  BNE     ERR3                ; R1= ADDRS OF TABL 1B
                                                            ; R6= 177 544 (RWADD + 2)

                                                ; TEST DEFERED ADDRESSING MODE
173042  011133                  MOV     (R1),@(R3)+         ; RWADD= DATA IN TABL 1         DM=3
173044  022135                  CMP     (R1)+,@(R5)+        ; RWADD= 177 400                DM=3
173046  001030                  BNE     ERR3                ; R1= ADDRS OF TABL 1C
                                                            ; R3 + R5= ADDRS OF TABL 3A

173050  020305                  CMP     R3,R5               ; ARE R3 AND R5 EQUAL           DM=0
173052  001026                  BNE     ERR3

173054  022705  173614          CMP     #TBL3A,R5           ; IS THE REGISTER VALUE CORRECT
173060  001023                  BNE     ERR3

173062  011153                  MOV     (R1),@-(R3)         ; RWADD= DATA IN TABL 1C        DM=5
173064  022155                  CMP     (R1)+,@-(R5)        ; RWADD= 000 377                DM=5
173066  001020                  BNE     ERR3                ; R1= ADDRS OF TABL 2
                                                            ; R3 + R5 = ADDRS OF TABL 3
```

```
     58 173070  020102                       CMP     R1,R2                   ; HAS R1 INCREMENTED CORRECTLY?
     59 173072  001016                       BNE     ERR3                    ; R2= ADDRS OF TABL 2
     60
     61 173074  010405                       MOV     R4,R5                   ; MAKE R5 = RWADD 177 5422
     62 173076  011114                       MOV     (R1),(R4)               ; RWADD= DATA IN TABL 2 DM=1
     63 173100  022115                       CMP     (R1)+,(R5)              ; RWADD= 052 525                    DM=1
     64 173102  001012                       BNE     ERR3                    ; R1= ADDRS OF TABL 2A
     65                                                                       ; R4 + R5 = ADDRS OF RWADD
     66
     67                                                              ; AUTOINCREMENT/AUTODECREMENT
     68
     69 173104  011124                       MOV     (R1),(R4)+              ; RWADD= DATA IN TABL 2A        DM=2
     70 173106  022125                       CMP     (R1)+,(R5)+             ; RWADD= 125 252                DM=2
     71 173110  001007                       BNE     ERR3                    ; R1= ADDRS OF TABL 2
     72                                                                       ; R4 + R5 = 177 544 (RWADD +2)
     73
     74 173112  020405                       CMP     R4,R5                   ; ARE THE TWO REGISTERS EQUAL?  DM=0
     75 173114  001005                       BNE     ERR3
     76
     77 173116  020604                       CMP     R6,R4                   ; R6= 177 544
     78 173120  001003                       BNE     ERR3                    ; CHECK THAT R4 IS CORRECT
     79
     80 173122  011145                       MOV     (R1),-(R5)              ; RWADD= DATA IN TABL 2B        DM=4
     81 173124  022144                       CMP     (R1)+,-(R4)             ; RWADD= 000 377                DM=4
     82 173126  001402                       BEQ     TEST4                   ; R1= ADDRS OF TABL 2C
     83                                                                       ; BRANCH TO NEXT TEST IF NO ERRORS.
     84
     85 173130  004067  000476        ERR3:  JSR     R0,CP3ERR               ; ERROR IN TEST 7
     86                                                                       ; GO TO CPU 2 ERROR.
     87                                                                       ; R0 WILL CONTAIN A POINTER TO HERE.
```

```
                        .SBTTL  CPU TEST 4
                        ;;*****************************************************************
                        ;;*****************************************************************
                        ;* TEST 4
                        ;* THIS TEST WILL CHECK THE MOVE AND COMPARE INSTRUCTIONS WITH A
                        ;* MIXTURE OF SOURCE MODES AND THE DESTINATION MODES WILL BE
                        ;* 37, 67, 77.
                        ;*
                        ;* THE READ WRITE ADDRESS REGISTER WILL BE USED IN THIS TEST.
                        ;*
                        ;* SM=0 SM=1 SM=R6/R7
                        ;*      THE SOURCE MODES ARE A MIXTURE OF PREVIOUSLY TEST ADDRESS MODES.
                        ;*
                        ;* TEST THE PC ADDRESSING MODES.
                        ;*
                        ;* DM=37 (@#A)
                        ;*      ABSOLUTE ADDRESSING
                        ;*      THE LOCATION FOLLOWING THE INSTRUCTION IS THE ADDRESS OF
                        ;*      THE OPERAND.
                        ;*
                        ;* SM=67 (A)
                        ;*      RELATIVE ADDRESS
                        ;*      THE LOCATION FOLLOWING THE INSTRUCTION IS USED AS A INDEX
                        ;*      TO LOCATE THE SELECTED LOCATION
                        ;*
                        ;* DM=77 OPR @X(PC)
                        ;*      RELATIVE DEFERED ADDRESSING
                        ;*      POINTS TO THE ADDRESS OF THE ADDRESS
                        ;*
                        ;* REGISTER VALUES
                        ;* INPUT REGISTERS
                        ;* R0= ADDRESS OF TABL 1B
                        ;* R1= RWADD ADDRESS
                        ;* R2= ADDRESS OF CHOS
                        ;* R3= 125 252
                        ;* R4= ADDRESS OF TABL 3
                        ;* R5= 052 525
                        ;* R6= RWADD ADDRESS MINUS 4
                        ;* RWADD=
                        ;*
                        ;* OUTPUT
                        ;* R0= ADDRESS OF TABL 1B
                        ;* R1= RWADD ADDRESS
                        ;* R2= ADDRESS OF CHOS
                        ;* R3= 125 252
                        ;* R4= ADDRESS OF TABL 3
                        ;* R5= 052 525
                        ;* R6= RWADD ADDRESS MINUS 4
                        ;* RWADD=
                        ;*
                        ;;*****************************************************************
```

```
 1 173134                          TEST4:
 2 173134  012702  173566                MOV     #CHOS1,R2          ; R3= ADDRS OF TABL 2   SM=27    DM=0
 3 173140  012203                        MOV     (R2)+,R3           ; R4= ADDRS OF TABL 3   SM=2     DM=0
 4 173142  012204                        MOV     (R2)+,R4           ; R5= ADDRS OF RWADD    SM=2     DM=0
 5 173144  011405                        MOV     (R4),R5            ; R2= ADDRS OF TBL1     SM=1     DM=0
 6
 7 173146  010637  177522                MOV     R6,@#RWADD         ; RWADD= 177 544        SM=0     DM=37
 8 173152  020615                        CMP     R6,(R5)            ; R5= RWADD ADDRS       SM=0     DM=1
 9 173154  001011                        BNE     ERR4
10
11 173156  011267  004340                MOV     (R2),RWADD         ; RWADD= DATA IN TBL1   SM=1     DM=67
12 173162  021215                        CMP     (R2),(R5)          ; R2= ADDRS OF TABL 1   SM=1     DM=1
13 173164  001005                        BNE     ERR4               ; RWADD= 125 252
14
15 173166  012777  052525  000416        MOV     #052525,@TBL3      ; RWADD= 052 525        SM=27    DM=77
16 173174  021315                        CMP     (R3),(R5)          ; R3= ADDRS OF TABL 2
17 173176  001402                        BEQ     TEST5              ; IF NO ERRORS GO TO TEST 5
18
19 173200  004067  000426          ERR4: JSR     R0,CP3ERR          ; ERROR IN TEST 4
20                                                                   ; GO TO CPU TEST 2 ERROR.
21                                                                   ; R0 WILL CONTAIN A POINTER TO HERE.
22
```

```
                        .SBTTL  CPU TEST 5
                        ;;*****************************************************************
                        ;;*****************************************************************
                        ;*
                        ;* TEST 5
                        ;* TEST THE SINGLE OPERANDS INSTRUCTIONS USING THE REGISTER
                        ;* ADDRESSING MODES 1 TO 7.
                        ;*
                        ;* REGISTER VALUES
                        ;* INPUT REGISTERS
                        ;* R0= ADDRESS OF TBL1
                        ;* R1= RWADD ADDRESS
                        ;* R2= ADDRESS OF CHOS
                        ;* R3= 125 252
                        ;* R4= ADDRESS OF TBL3
                        ;* R5= 052 525
                        ;* R6= RWADD ADDRESS MINUS 4
                        ;* RWADD=
                        ;*
                        ;* OUTPUT REGISTERS
                        ;* R0= ADDRESS OF TBL1A
                        ;* R1= RWADD ADDRESS
                        ;* R2= ADDRESS OF CHOS
                        ;* R3= 125 252
                        ;* R4= ADDRESS OF TABL 3
                        ;* R5= 052 525
                        ;* R6= RWADD ADDRESS MINUS 4
                        ;* RWADD=
                        ;*
                        ;;*****************************************************************
                        ;;*****************************************************************
```

```
 1 173204                          TEST5:
 2 173204  012702  173570                  MOV     #CHOS2,R2       ; R6= RWADD+2 SETUP IN PREVIOUS TEST.
 3 173210  011203                          MOV     (R2),R3         ; R4= RWADD ADDRESS (177 542)
 4 173212  011304                          MOV     (R3),R4         ; R3= ADDRESS OF TABL 3
 5                                                                  ; R2= ADDRESS OF CHOS 2
 6
 7                                                                  ; RWADD =       NZVC
 8 173214  005014                          CLR     (R4)            ; 000 000       0100            DM=1
 9 173216  005124                          COM     (R4)+           ; 177 777       1001            DM=2
10 173220  005444                          NEG     -(R4)           ; 000 001       0001            DM=4
11 173222  006033                          ROR     @(R3)+          ; 100 000       1001            DM=3
12 173224  005553                          ADC     @-(R3)          ; 100 001       1000            DM=5
13 173226  000261                          SEC                     ;               1001
14
15 173230  006172  000022                  ROL     @22(R2)         ; 000 003       0011            DM=7
16 173234  006266  177776                  ASR     -2(R6)          ; 000 001       0011            DM=6
17 173240  005614                          SBC     (R4)            ; 000 000       0100            DM=1
18 173242  006324                          ASL     (R4)+           ; 000 000       0100            DM=2
19 173244  005244                          INC     -(R4)           ; 000 001       0000            DM=4
20 173246  005333                          DEC     @(R3)+          ; 000 000       0100            DM=3
21 173250  005753                          TST     @-(R3)          ; 000 000       0100            DM=5
22 173252  001402                          BEQ     TEST5A          ; DO THE BYE TEST
23 173254  004067  000352          ERR5:   JSR     R0,CP3ERR               ; ERROR IN TEST 5
24                                                                  ; GO TO CPU TEST 2 ERROR.
25                                                                  ; R0 WILL CONTAIN A POINTER TO HERE.
26
27 173260  012714  177777          TEST5A: MOV     #-1,(R4)        ; 177 777       1000            DM=1
28 173264  105014                          CLRB    (R4)            ; 177 400       0100            DM=1
29 173266  105124                          COMB    (R4)+           ; 177 777       1001            DM=2
30 173270  105644                          SBCB    -(R4)           ; 177 776       1000            DM=4
31 173272  105233                          INCB    @(R3)+          ; 177 777       1000            DM=3
32 173274  105453                          NEGB    @-(R3)          ; 177 401       0001            DM=5
33 173276  105372  000022                  DECB    @22(R2)         ; 177 400       0101            DM=7
34 173302  000366  177776                  SWAB    -2(R6)          ; 000 377       1000            DM=6
35 173306  106314                          ASLB    (R4)            ; 000 376       1001            DM=1
36 173310  106024                          RORB    (R4)+           ; 000 377       1010            DM=2
37 173312  106244                          ASRB    -(R4)           ; 000 377       1001            DM=4
38 173314  106133                          ROLB    @(R3)+          ; 000 377       1001            DM=3
39 173316  105553                          ADCB    @-(R3)          ; 000 000       0101            DM=5
40 173320  105724                          TSTB    (R4)+           ; 000 000       0100
41 173322  001002                          BNE     ERR5A           ; TEST THE FIRST PART OF RWADD
42 173324  105724                          TSTB    (R4)+           ; TEST THE HIGH BYTE OF RWADD
43 173326  001402                          BEQ     TEST6           ; GO TO TEST 6
44
45 173330  004067  000276          ERR5A:  JSR     R0,CP3ERR               ; ERROR IN TEST 5A
46                                                                  ; GO TO CPU 2 ERROR.
47                                                                  ; R0 WILL CONTAIN A POINTER TO HERE.
```

```
                                .SBTTL  CPU TEST 6
                                ;* TEST 6
                                ;* THIS TEST WILL CHECK THE ADDRESSING MODES THAT WERE NOT CHECKED
                                ;* IN THE PREVIOUS TESTS USING [ MOV, MOVB, CMP, CMPB ]
                                ;* REGISTER VALUES
                                ;* INPUT REGISTERS
                                ;* R0= ADDRESS OF TBL1A
                                ;* R1= RWADD ADDRESS
                                ;* R2= ADDRESS OF CHOS
                                ;* R3= 125 252
                                ;* R4= ADDRESS OF TABL 3
                                ;* R5= 052 525
                                ;* R6= RWADD ADDRESS MINUS 4
                                ;* RWADD =
                                ;* OUTPUT
                                ;* R0=
                                ;* R1=
                                ;* R2=
                                ;* R3=
                                ;* R4=
                                ;* R5=
                                ;* R6=
                                ;* RWADD =
173334                          TEST6:
173334  012701  173570                  MOV     #CHOS2,R1               ; R1= CHOS      SM=27   DM=0
173340  011104                          MOV     (R1),R4                 ; R2= TBL2      SM=1    DM=0
173342  011403                  MOV     (R4),R3                 ; R3= RWADD     SM=1    DM=0
173344  014102                  MOV     -(R1),R2                ; R4= TBL3      SM=1    DM=0
173346  014105                  MOV     -(R1),R5                ; R5= TBL1      SM=1    DM=0
173350  005013                  CLR     (R3)                    ; RWADD = 000 000
173352  113134                  MOVB    @(R1)+,@(R4)+           ; RWADD= 000 252        SM=3    DM=3
173354  121513                  CMPB    (R5),(R3)               ; R1= CHOS1             SM=1    DM=1
173356  001021                  BNE     ERR6                    ; R4= TBL3A
173360  111305                  MOVB    (R3),R5         ; R5= 177 652           SM=1    DM=0

173362  113113                  MOVB    @(R1)+,(R3)             ; RWADD= 000 125        SM=3    DM=1
173364  121254                  CMPB    (R2),@-(R4)             ; R1= CHOS2             SM=1    DM=5
173366  001015                  BNE     ERR6                    ; R4= TBL3

173370  011302                  MOV     (R3),R2                 ; R2= 000 125           SM= 1   DM=0
173372  022705                  CMP     (PC)+,R5                ; CHECK R5 = 177 652    SM=27   DM=0
173374  177652                  .WORD   177652
173376  001011                  BNE     ERR6
173400  022702  000125          CMP     #000125,R2              ; CHECK R2 = 000 125    SM= 27  DM=0
173404  001006                  BNE     ERR6
173406  113104                  MOVB    @(R1)+,R4               ; R4= 000 142           SM= 3   DM=0
```

CPU TEST 6

```
     58 173410  120304                       CMPB    R3,R4                ; R3= 177 542          SM=0    DM=0
     59 173412  001003                       BNE     ERR6                 ; R1= TBL1
     60
     61 173414  112103                       MOVB    (R1)+,R3             ; R1= TBL1+1           SM= 2   DM=0
     62 173416  120503                       CMPB    R5,R3                ; R3= 177 652          SM=0    DM=0
     63 173420  001402                       BEQ     TEST7                ; BRANCH IF NO ERRORS IN THIS TEST
     64
     65 173422  004067  000204        ERR6:  JSR     R0,CP3ERR            ; IF TEST 6 IN ERROR GO TO CP3 ERROR
     66                                                                   ; R0 WILL POINT TO TEST 6 THAT CAUSED
     67                                                                   ; THE ERROR.
```

```
                        .SBTTL  CPU TEST 7
                        ; TEST 7
                        ; THIS TEST WILL CHECK THE BIT TEST, BIT SET, BIT CLEAR.
                        ; THE INSTRUCTIONS WILL BE CHECKED IN THE FOLLOWING MANER:
                        ;
                        ; BIT TEST, BIT SET, BIT CLEAR          DESTINATION MODE = 0    WORD
                        ;
                        ; BIT TEST, BIT SET, BIT CLEAR          DESTINATION MODE NOT = 0        WORD
                        ;
                        ; BIT TEST, BIT SET, BIT CLEAR          DESTINATION MODE = 0    BYTE
                        ;
                        ; BIT TEST, BIT SET, BIT CLEAR          DESTINATION MODE NOT = 0        BYTE
                        ;
                        ;* REGISTER VALUES
                        ;* INPUT        TEST 7 SET UP REQUIRED REGISTERS
                        ;* R0= ADDRESS OF CHOS1
                        ;* R1= ADDRESS OF TBL2
                        ;* R2= ADDRESS OF TBL3
                        ;* R3= RWADD ADDRESS
                        ;* R4= 177 777
                        ;*
                        ;* OUTPUT
                        ;* R0=
                        ;* R1=
                        ;* R2=
                        ;* R3=
                        ;* R4=
                        ;* R5=
                        ;* R6=
                        ;*
                        .ENABLE LSB
173426  012700  173564  TEST7:  MOV     #CHOS,R0        ; R0= 177 777
173432  012001                  MOV     (R0)+,R1        ; R1= ADDRESS OF TBL1
173434  012002                  MOV     (R0)+,R2        ; R2= ADDRESS OF TBL2
173436  012003                  MOV     (R0)+,R3        ; R3= ADDRESS OF TBL3
173440  011304                  MOV     (R3),R4         ; R4= RWADD ADDRESS
173442  012700  177777          MOV     #-1,R0
173446  000277                  SCC                     ; SET ALL CONDITION CODES(CC) TO 1
173450  050005                  BIS     R0,R5           ; R5= 1777 777          SM=0    DM=0
173452  100042                  BPL     ERR7            ; CC BIT N=1
                                                        ; R0= 177 777
173454  041205                  BIC     (R2),R5         ; CLEAR BITS 052 525    SM=1    DM=0
173456  032205                  BIT     (R2)+,R5        ; R5= 125 252           SM=1    DM=0
173460  001037                  BNE     ERR7            ; R1= TBL1 / R2= TBL2A
                                                        ; TEST BITS 052 525
173462  050033                  BIS     R0,@(R3)+       ; SET RWADD = 177 777   SM=0    DM=3
173464  100035                  BPL     ERR7            ; WAS CC BIT N=1

173466  042153                  BIC     (R1)+,@-(R3)    ; CLEAR 125 252         SM=2    DM=5
```

```
     58 173470  100433                        BMI     ERR7            ; RWADD = 052 525
     59                                                                 ; R1= TBL 1A
     61 173472  031214                        BIT     (R2),(R4)       ; TEST BIT 125 252      SM=2    DM=2
     62 173474  001031                        BNE     ERR7            ; NO BIT SHOULD BE SET
     63
     64 173476  022137  177522                CMP     (R1)+,@#RWADD   ; CHECK RWADD= 052525
     65 173502  001026                        BNE     ERR7
     66
     67 173504  022205                        CMP     (R2)+,R5        ; TEST R5 WITH 125 377
     68 173506  001024                        BNE     ERR7            ; R2= TBL2B
     69
     70 173510  150005                        BISB    R0,R5           ; R5= 125 777           SM=1    DM=0
     71 173512  100022                        BPL     ERR7            ; R0 = 377
     72
     73 173514  144205                        BICB    -(R2),R5        ; CLEAR BITS 277        SM=1    DM=0
     74 173516  100420                        BMI     ERR7            ; R5= 125 125
     75                                                                 ; R2= TABLE 2A + 1
     76
     77 173520  131205                        BITB    (R2),R5         ; R2= 252
     78 173522  001016                        BNE     ERR7            ; R5= 125 125
     79                                                                 ; R2= TBL 1B
     80
     81 173524  000305                        SWAB    R5              ; R5= 052 652
     82 173526  100014                        BPL     ERR7
     83
     84 173530  130514                        BITB    R5,(R4)         ; R5 LOW BYTE= 252      SM=0    DM=1
     85 173532  001012                        BNE     ERR7            ; RWADD LOW BYTE= 125
     86
     87 173534  150514                        BISB    R5,(R4)         ; RWADD LOW BYTE= 377   SM=0    DM=1
     88 173536  100010                        BPL     ERR7            ; RWADD= 052 777
     89
     90 173540  000314                        SWAB    (R4)            ; RWADD = 177525        SM= 1
     91 173542  100406                        BMI     ERR7            ;
     92 173544  131233                        BITB    (R2),@(R3)+     ; TEST BIT IN RWADD LOW BYTE    SM=2    DM=3
     93 173546  001004                        BNE     ERR7            ; LOW BYTE RWADD = 125
     94                                                                 ; LOW BYTE R2    = 252
     95
     96 173550  144153                        BICB    -(R1),@-(R3)    ; CLEAR LOW BYTE RWADD          SM= 5   DM= 3
     97 173552  001002                        BNE     ERR7            ; RWADD = 1774000
     98                                                                 ; CLEAR BYTE= 125
     99                                                                 ; R1= TBL 1A + 1
    100
    101 173554  021214                        CMP     (R2),(R4)       ; DOES RWADD = 177400
    102 173556  001017                        BNE     NXTTST          ; BRANCH IF NO ERRORS
    103                                                                ; R0= 177 777   RWADD= 177 400
    104                                                                ; R1= TBL 1A + 1         R2= TBL 2A + 1
    105                                                                ; R3= TBL 3      R4= RWADD       R5= 052 652
    106                                                    ; ***************************************************
    107 173560  004067  000046        ERR7:   JSR     R0,CP3ERR               ; ERROR IN TEST 7 R0 WILL POINT
    108                                                                        ; USER TO HERE . ERROR IN TEST 7
    109                               .DSABL LSB
```

```
                                                                ; TABLES USED IN THIS PAGE.
                                                                ;TBL1
 173564  173572          CHOS:   TBL1
 173566  173602          CHOS1:  TBL2
 173570  173612          CHOS2:  TBL3

 173572  125252          TBL1:   125252
 173574  052525          TBL1A:  052525
 173576  177400          TBL1B:  177400
 173600  000377          TBL1C:  000377

 173602  052525          TBL2:   052525
 173604  125252          TBL2A:  125252
 173606  000377          TBL2B:  000377
 173610  177400          TBL2C:  177400

 173612  177522          TBL3:   RWADD
 173614  173612          TBL3A:  TBL3
```

```
                                ;* NEXT TEST ROUTINE
                                ;: SET UP R4 AND R5 WITH THE NEW PAGE AND PC.
                                ;* PROCEED TO NEXT CPU TEST.

 173616  012705  173000     NXTTST: MOV     #TEST8,R5               ;GO TO TEST 8.
 173622  012704  003406             MOV     #PAGE6,R4               ; SET UP CORRECT PAGE
 173626  000167  000134             JMP     CPAGE5                  ; GO CHANGE PAGE


                                .SBTTL  CPU ERROR HALT THREE.

 173632  000000             CP3ERR: HALT                            ; ERROR OCCURED IN THE CPU TESTS
                                                                     ; IN THIS ROM PAGE.
                                                                     ; R0 = POINTER TO ERROR ROUTINE.
                                                                     ; R1 TO R5 ARE ERROR DATA INDICATORS.

 173634  012705  173000             MOV     #START,R5               ; ODT PROCEED WILL RESTART THE ROM CODE
 173640  012704  000400             MOV     #PAGE0,R4               ; NEW PAGE
 173644  000450                     BR      CPAGE5                  ; AT START AND PAGE 0

                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE FIVE.
                                ;* CHANGE PAGE ROUTINE FOR THIS PAGE.
                                ;*
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;*
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;*
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*

                                                        ;FILL REMAINING ROM LOCATIONS WITH HALTS
 173646  000000                 .WORD   000000
 173650  000000                 .WORD   000000
 173652  000000                 .WORD   000000
 173654  000000                 .WORD   000000
 173656  000000                 .WORD   000000
 173660  000000                 .WORD   000000
 173662  000000                 .WORD   000000
 173664  000000                 .WORD   000000
 173666  000000                 .WORD   000000
 173670  000000                 .WORD   000000
```

CHANGE PAGE ROUTINE FOR PAGE FIVE.

```
 59 173672  000000                    .WORD   000000
 60 173674  000000                    .WORD   000000
 61 173676  000000                    .WORD   000000
 62 173700  000000                    .WORD   000000
 63 173702  000000                    .WORD   000000
 64 173704  000000                    .WORD   000000
 65 173706  000000                    .WORD   000000
 66 173710  000000                    .WORD   000000
 67 173712  000000                    .WORD   000000
 68 173714  000000                    .WORD   000000
 69 173716  000000                    .WORD   000000
 70 173720  000000                    .WORD   000000
 71 173722  000000                    .WORD   000000
 72 173724  000000                    .WORD   000000
 73 173726  000000                    .WORD   000000
 74 173730  000000                    .WORD   000000
 75 173732  000000                    .WORD   000000
 76 173734  000000                    .WORD   000000
 77 173736  000000                    .WORD   000000
 78 173740  000000                    .WORD   000000
 79 173742  000000                    .WORD   000000
 80 173744  000000                    .WORD   000000
 81 173746  000000                    .WORD   000000
 82 173750  000000                    .WORD   000000
 83 173752  000000                    .WORD   000000
 84 173754  000000                    .WORD   000000
 85 173756  000000                    .WORD   000000
 86 173760  000000                    .WORD   000000
 87 173762  000000                    .WORD   000000
 88 173764  000000                    .WORD   000000
 89         173766                    .=      173766
 91 173766  010437  177520  CPAGE5:   MOV     R4,@#PAGELO             ; LOAD NEW PAGE WORD.
 92 173772  010507                    MOV     R5,PC                   ; JAM NEW PC.
 93
 94 173774  000000                    .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
 95
 96 173776  000000                    .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
 97                                                                   ; BY ROM BLASTING PROGRAM.
 98
 99         000001'                   .END
```

TABLE OF CONTENTS

1-	9	CPU TESTS AND CONSOL TERMINAL TEST.
2-	1	BASIC DEFINITIONS
3-	1	LOCAL DATA
4-	1	SPECIAL EQUALITIES FOR BVD11-AA
5-	1	CPU TEST 8
7-	1	CPU TEST 9
8-	1	CPU TEST 10
9-	1	TEST DATA FOR ROMD
11-	1	CPU ERROR HALT FOUR.
11-	17	END OF CPU TESTS.
12-	1	CONSOL TERMINAL TEST
15-	1	CHANGE PAGE ROUTINE FOR PAGE SEVEN.


```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS
                                
173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550
                                
000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          . = 173000
```

```
                                .SBTTL  CPU TEST 8
                                ;* TEST 8
                                ;* THIS TEST WILL CHECK THE ADD INSTRUCTION FOR ITS ABILITY TO
                                ;* PERFORM THE CORRECT ARITHRMITIC AND SET THE CORRECT CONDITION CODES.
                                                ; ROM PAGE IS PAGE6. 003406 IN PAGE CONTROL REGISTER.
                                                ; DIAGNOSTIC LIGHT DISPLAY = 1.
173000                          TEST8:
173000  012701  173234                  MOV     #TEST6,R1               ; R1= ADDRESS OF TEST 5
173004  011104                          MOV     (R1),R4                 ; R3 = 052 525
173006  014103                          MOV     -(R1),R3                ; R4 = 025 252

173010  061104                          ADD     (R1),R4                 ; R1 = + 052 252
173012  101425                          BLOS    ERR8                    ; R4 = + 025 252
173014  003424                          BLE     ERR8                    ; --------------
                                                                        ; R4 = + 077 777
                                                                        ; N=0  Z=0  V=0  C=0    CC=00

173016  061104                          ADD     (R1),R4                 ; R1 = + 052 525
173020  101422                          BLOS    ERR8                    ; R4 = + 077 777
173022  002421                          BLT     ERR8                    ; --------------
                                                                        ; R4 = - 152 524
                                                                        ; N=1  Z=0  V=1  C=0    CC=12

173024  061104                          ADD     (R1),R4                 ; R1 = + 052 525
173026  003417                          BLE     ERR8                    ; R4 = - 152 524
                                                                        ; --------------
                                                                        ; R4 = + 025 251
                                                                        ; N=0  Z=0  V=0  C=1    CC=01

173030  010451                  MOV     R4,@-(R1)               ; RWADD= 052 251
173032  011104                  MOV     (R1),R4                 ; R4   = RWADD

173034  064114                  ADD     -(R1),(R4)              ; R1 = - 125 252
173036  101413                  BLOS    ERR8                    ; R4 = + 025 251
173040  002012                  BGE     ERR8                    ; --------------
                                                                ; R4 =  152 523
                                                                ; N=1  Z=0  V=0  C=0    CC=10

173042  064114                  ADD     -(R1),(R4)              ; R1 = - 152 525
173044  101010                  BHI     ERR8                    ; R4 = - 152 523
173046  002007                  BGE     ERR8                    ; --------------
                                                                ; R4 =  125 250
                                                                ; N=1  Z=0  V=0  C=1    CC=11

173050  061114                  ADD     (R1),(R4)               ; R1 = - 152 525
173052  101005                  BHI     ERR8                    ; R4 = - 125 250
173054  002004                  BGE     ERR8                    ; --------------
                                                                ; R4 =  077 775
                                                                ; N=0  Z=0  V=1  C=1    CC=03
```

CPU TEST 8

```
      1 173056  064114                        ADD     -(R1),(R4)              ; R1 = - 100 003
      2 173060  002402                        BLT     ERR8                    ; R4 = + 077 775
      3 173062  003001                        BGT     ERR8                    ; --------------
      4 173064  001402                        BEQ     TEST9                   ; R4 =  000 000
      5                                                                       ; N=0  Z=1  V=0  C=1    CC=05
      6
      7 173066  004067  000144          ERR8: JSR     R0,CP4ERR               ; ERROR IN TEST 8
      8                                                                       ; CALL CPU ERR 3 ON THIS PAGE.
      9                                                                       ; R0 WILL POINTER TO THIS ERROR.
```

```
                        .SBTTL  CPU TEST 9
                        ;******************************************************************************
                        ;******************************************************************************
                        ;* TEST 9
                        ;* THIS TEST WILL TEST THE SUBTRACK INSTRUCTION.
                        ;******************************************************************************
                        ;******************************************************************************
173072                  TEST9:
173072  012701  173234          MOV     #TESTD6,R1              ; R1 = ADDRESS OF TESTD5
173076  014104                  MOV     -(R1),R4                ; R4 =+ 052 525
173100  161104                  SUB     (R1),R4                 ; R4 =+ 052 525
173102  002427                  BLT     ERR9                    ; R1 =+ 052 525
173104  003026                  BGT     ERR9                    ; -------------
173106  103425                  BLO     ERR9                    ; R4 =+ 000 000
                                                                ; N=0  Z=1  V=0  C=0    CC=04
173110  161104                  SUB     (R1),R4                 ; R4 =+ 000 000
173112  003023                  BGT     ERR9                    ; R1 =+ 052 525
173114  101022                  BHI     ERR9                    ; -------------
                                                                ; R4 =- 125 253
                                                                ; N=1  Z=0  V=0  C=1    CC=11
173116  161104                  SUB     (R1),R4                 ; R4 =- 125 253
173120  101420                  BLOS    ERR9                    ; R1 =+ 052 525
173122  002017                  BGE     ERR9                    ; -------------
                                                                ; R4 =+ 052 526
                                                                ; N=0  Z=0  V=1  C=0    CC=02
173124  010451                  MOV     R4,@-(R1)               ; RWADD = 052 526
173126  011104                  MOV     (R1),R4                 ; R4    = RWADD
173130  164114                  SUB     -(R1),(R4)              ; R4 =+ 052 526
173132  003413                  BLE     ERR9                    ; R1 =- 125 252
173134  101012                  BHI     ERR9                    ; ------------
                                                                ; R1 =- 125 254
                                                                ; N=1  Z=0  V=1  C=1    CC=13
173136  161114                  SUB     (R1),(R4)               ; R4 =- 125 254
173140  103410                  BCS     ERR9                    ; R1 =- 125 252
173142  102407                  BVS     ERR9                    ; -------------
                                                                ; R4 =+ 000 002
                                                                ; N=0  Z=0  V=0  C=0    CC=00
173144  005741                  TST     -(R1)                   ; DECREMENT POINTER
173146  164114                  SUB     -(R1),(R4)              ; R4 =+ 000 002
173150  103004                  BHIS    ERR9                    ; R1 =- 100 003
173152  003403                  BLE     ERR9                    ; --------------
                                                                ; R4 =  077 777
                                                                ; N=0  Z=0  V=0  C=1    CC=01
173154  022714  077777          CMP     #077777,(R4)            ; WAS THE RESULT CORRECT?
173160  001402                  BEQ     TEST10                  ; DO BRANCH IF CORRECT.
```

```
     58
     59 173162  004067  000050          ERR9:   JSR     R0,CP4ERR               ; ERROR IN TEST 9
     60                                                                        ; R0 WILL POINT TO SUBTEST
     61                                                                        ; CAUSING THE ERROR.
```

```
                        .SBTTL  CPU TEST 10
                        ;*************************************************************************
                        ;*************************************************************************
                        ;* TEST 10
                        ;* TEST THE SOB INSTRUCTION(SUBTRACT ONE AND BRANCH IF NOT = 0)
                        ;*
                        ;*
                        ;*
                        ;*
                        ;*************************************************************************
                        ;*************************************************************************

173166  012701  000002  TEST10: MOV     #2,R1                   ; R1 = 000 002
173172  010102                MOV     R1,R2                   ; R2 = 000 002

173174  005302          10$:    DEC     R2                      ; R2 IS A COUNTER USED TO VERFIY
173176  002407                  BLT     ERR10                   ; THE SOB INSTRUCTION IS EXECUTING
173200  005701                  TST     R1                      ; R1 IS THE REGISTER THE SOBINST.
173202  002405                  BLT     ERR10                   ; WILL SUBTRACT  FROM
173204  077105                  SOB     R1,10$                  ; THE FIRST PASS THROUGH NO BRANCH
173206  005701                  TST     R1                      ; BUT THE SECOND PASS WILL BRANCH
173210  001002                  BNE     ERR10                   ; BRANCH IF ERROR.
173212  000167  000040          JMP     ENDCPU                  ; SOB IS WORKING!

173216  004067  000014  ERR10:  JSR     R0,CP4ERR               ; SOB INSTRUCTION FAILED
                                                                ; R0 - POINTER TO CAUSE OF ERROR
                                                                ; R1 = TEST COUNTER
                                                                ; R2 - SOB SUBTRACT REGISTER
                                                                ; R3 = NOT USED
                                                                ; R4 = NOT USED
                                                                ; R5 = NOT USED
```

```
                                .SBTTL  TEST DATA FOR ROMD
                                ;******************************************************************************
                                ;******************************************************************************

173222  100003                  TESTD1: 100003

173224  152525                  TESTD2: 152525

173226  125252                  TESTD3: 125252

173230  177522                  TESTD4: RWADD

173232  052525                  TESTD5: 052525

173234  025252                  TESTD6: 025252
```

```
                                .SBTTL  CPU ERROR HALT FOUR.
                                ;*************************************************************************
                                ;*************************************************************************
                                ;*
                                ;* ERROR HALT FOR ROMD.
                                ;*
                                ;*************************************************************************
                                ;*************************************************************************

173236  000000                  CP4ERR: HALT
173240  005726                          TST     (SP)+           ; CORRECT STACK
173242  012704  000400                  MOV     #PAGE0,R4       ; RETURN TO PAGE ZERO
173246  012705  173000                  MOV     #START,R5       ; RETURN ADDRESS
173252  000167  000510                  JMP     CPAGE7


                                .SBTTL  END OF CPU TESTS.
                                ; *
                                ; *
                                ;* FINISHED WITH THE CPU HARDCORE TEST RETURN TO THE SWITCH CHECKING ROUTINE
                                ;* AND CHECK FOR SWITCH (BIT 1 ) TO TEST THE MEMORY.
                                ;*
                                ; *
                                ; *

173256  012700  000002          ENDCPU: MOV     #BIT1,R0                ; SETUP BIT TO BE TESTED
173262  012701  173554                  MOV     #TABLE+004,R1           ; R0 = CHECK BIT FOR MEMORY TEST
                                                                        ; R1 = ENTRY INTO TABLE

173266  012704  000400                  MOV     #PAGE0,R4               ; R4 = RETURN PAGE
173272  012705  173514                  MOV     #SWTEST,R5              ; R5 = ADDRESS TO RETURN TO
173276  000167  000464                  JMP     CPAGE7                  ; DO CHANGE PAGE
```

```
                        .SBTTL  CONSOL TERMINAL TEST
                        ;;*****************************************************************************
                        ;;*****************************************************************************
                        ;* TEST THE CONSOL TERMINAL.
                        ;* THIS TEST WILL INDICATE THE MEMORY SIZE (IF THE MEMORY TEST WAS RUN)
                        ;* FOUND IN THIS SYSTEM AND ASK THE USER TO ANSWER THE "START?" QUESTION.
                        ;* THIS IS A BASIC TEST OF THE TERMINAL.
                        ;*
                        ;* EXAMPLE:
                        ;*
                        ;* 16
                        ;* START?  "Y"
                        ;* "RT-11SJ     V02C-02
                        ;*  . "
                        ;*
                        ;*  TWO DIGIT MEMORY SIZE=      4   = 017 776   16  = 077 776
                        ;*                              8   = 037 776   20  = 117 776
                        ;*                              12  = 057 776   24  = 137 776
                        ;*                                              28  = 157 776
                        ;*                              ?? = NO EXPECTED MEMORY VALUE FOUND.
                        ;*
                        ;* CPU WILL TYPE                          START? (Y OR N)
                        ;*
                        ;* USER WILL TYPE "Y" TO BOOT SYSTEM.    "Y" = YES
                        ;* USER WILL TYPE "N" TO HALT            "N" = NO
                        ;* AND ENTER MICRO CODE ODT.
                        ;*
                        ;* ANY OTHER CHARACTER WILL REASK THE START QUESTION..
                        ;*
                        ;* REGISTER VALUES
                        ;* INPUT
                        ;* R0= IS THE BIT THAT MATCHED A SWITCH SETTING.
                        ;* R1= NEXT ENTRY INTO TABLE.
                        ;* R2= RESULT OF SOME TEST, NOT USED IN SWITCH CHECK.
                        ;* R3= MEMORY UPPER LIMIT VALUE FROM "MMTEST".
                        ;* R4= PAGE CONTROL REGISTER SETTING.
                        ;* R5= ADDRESS OF ROUTINE TO BE EXECUTED.
                        ;*
                        ;* OUTPUT VALUES
                        ;* R0= BIT FOR NEXT SWITCH TEST(ROM AREA BOOT).
                        ;* R1= ENTRY INTO TABLE.
                        ;* R2= RESULT OF SOME TEST, NOT USED IN SWITCH CHECK.
                        ;* R3= RESULT OF SOME TEST, NOT USED IN SWITCH CHECK.
                        ;* R4= PAGE CONTROL REGISTER SETTING.
                        ;* R5= ADDRESS OF TERMINAL TEST ROUTINE.
                        ;* R6= RWADD + 2 (177 524)
                        ;*
                        ;;*****************************************************************************
                        ;;*****************************************************************************
```

```
                                                ; ROM PAGE IS PAGE10. 003406 IN PAGE CONTROL REGISTER.
173302                          VTTEST:         ; DIAGNOSTIC LIGHT DISPLAY = '3'
173302  112737  000014  177524          MOVB    #THREE,@#LIGHTS         ; LIGHTS = 3 FOR DISPLAY TEST.
173310  012706  000004                  MOV     #4,SP                   ; LOCATE TOP OF MEMORY
173314  012726  173330                  MOV     #FOUND1,(SP)+           ; NON- EXISTANT MEMORY TRAP TO FOUND1
173320  012716  000340                  MOV     #340,(SP)               ; PSW OF INTERRUPT RETURN

173324  011626                  10$:    MOV     (SP),(SP)+              ; TEST FOR NON-EXISTANT MEMORY
173326  000776                  10$     BR      10$                     ; BRANCH IF STILL MEMORY
                                                                        ; TRAP IF NO MEMORY.
173330  005726                  FOUND1: TST     (SP)+                   ; ADD 2 TO SP FOR LAST MEMORY LOCATION
173332  010603                  MOV     SP,R3                   ; MEMORY HIGH LIMIT IN R3
173334  012702  157776          MOV     #157776,R2              ; R2 = VALUE FOR MAX. SYSTEM 28K
                                                                ; R3=UPPER MEMORY LIMIT SET IN MEM TEST.
173340  012704  173554          MOV     #TABLE5,R4              ; R4 = ASCII PRINT BUFFER
173344  004767  000154          5$:     JSR     PC,PRINT                ; PRINT A SPACE,LF,CR
173350  105714                  TSTB    (R4)                    ; LOOK FOR END OF PRINT BUFFER
173352  001374                  BNE     5$                      ; BRANCH BACK IF MORE TO DO.
173354  012704  173561          MOV     #TABLE6,R4              ; R4 = ASCII BUFFER OF MEMORY SIZES.
173360  020203                  10$:    CMP     R2,R3           ; R3= UPPER MEM LIMIT OF SYS
173362  001404                  BEQ     20$                     ; R2= VALUE USED COMPARING 4K MEM BANKS
173364  005724                  TST     (R4)+                   ; WHEN MATCH GOTO 20$
173366  162702  020000          SUB     #020000,R2              ; NO MATCH GOTO NEXT LOWER 4K BANK.
173372  103372                  BCC     10$                     ; IF CARRY SET AFTER 'SUB' THEN NO 4K
                                                                ; MEM. VALUE FOUND . PRINT ??.
173374  004767  000124          20$:    JSR     PC,PRINT                ; R4 = ADDRS OF MEM SIZE TO BE PRINTED
173400  004767  000120                  JSR     PC,PRINT                ; MEM SIZE IS TWO CHAR NO TERMINATOR

173404  012704  173601          25$:    MOV     #TABLE7,R4              ; ASCII BUFFER FOR CR,LF,'START'
173410  004767  000110          30$:    JSR     PC,PRINT                ; PRINT THE MESSAGE
173414  105714                  TSTB    (R4)                    ; TERMINATOR FOUND?
173416  001374                  BNE     30$                     ; BRANCH IF NO TERMINATOR FOUND.

                                ; KEYBOARD ROUTINE
                                ; DIAGNOSTIC LIGHTS = 4
173420  112737  000013  177524          MOVB    #FOUR,@#LIGHTS          ; SET LIGHTS TO A 4.
173426  105737  177560          40$:    TSTB    @#TKS                   ; CHECK FOR KEY STRUCK.
173432  100375                  BPL     40$                     ; LOOP LOOKING FOR KEY DONE.
                                                                ; IF NO KEY DONE FLAG WAIT HERE FOPEVER.
                                ; KEYBOARD FLAG SET
173434  012704  177562          MOV     #TKB,R4                 ; R4 = ADDRESS OF KEY BOARD BUFFER.
173440  111403                  MOVB    (R4),R3                 ; R3 = ASCII DATA FOR KEY.
173442  004767  000056          JSR     PC,PRINT                ; PRINT KEY TYPED IN STORED IN TKB
173446  142703  000240          BICB    #240,R3                 ; REMOVE PARITY FROM CHAR TYPED IN.
                                                                ; READ UPPER OR LOWER CASE CHAR

173452  012704  173613          MOV     #TABL10,R4              ; ASCII TABLE FOR Y AND N.
173456  122403                  CMPB    (R4)+,R3                ; CHECK IF IT IS A YES
173460  001015                  BNE     50$                     ; NOT A Y . DO BRANCH
```

```
     58 173462  012704  173554                MOV     #TABLE5,R4              ; R4 = ASCII PRINT BUFFER
     59
     60                                                                 ; CHARACTER IS A Y(YES) RETURN TO SWITCH
     61                                                                 ; CECKING ROUTINE FOR LOAD DEVICE.
     62 173466  004767  000032          45$:    JSR     PC,PRINT                ; PRINT A SPACE,LF,CR SO RT-11
     63                                                                         ; WILL START ON NEW LINE.
     64 173472  105714                          TSTB    (R4)                    ; LOOK FOR END OF PRINT BUFFER
     65 173474  001374                          BNE     45$                     ; BRANCH BACK IF MORE TO DO.
     66
     67                                                                 ; RETURN TO SWICH CHECKING ROUTINE.
     68                                                                         ; VALUE IN R1 IS CORRECT RETURN ADDRESS.
     69 173476  006300                          ASL     R0                      ; ROTATE R0 FOR CORRECT ENTRY.
     70 173500  012704  000400                  MOV     #PAGE0,R4               ; R4 = CHANGE TO PAGE 0.
     71 173504  012705  173514                  MOV     #SWTEST,R5              ; R5 = STARTING ADDRESS ON NEW PAGE.
     72 173510  000167  000252                  JMP     CPAGE7                  ; CHANGE PAGE
     73
     74 173514  121403                  50$:    CMPB    (R4),R3                 ; CHECK IF IT IS A NO
     75 173516  001332                          BNE     25$                     ; IF NOT A NO REASK START QUESTION.
     76
     77                                                                 ; **********************************************
     78 173520  000000                          HALT                            ; STOP ON A NO TYPED IN.
     79                                                                 ; **********************************************
     80 173522  000730                          BR      25$                     ; REASK START QUESTION.
     81
     82                                                                 ; PRINT ROUTINE
     83 173524  005005                  PRINT:  CLR     R5                      ; WATCH DOG TIMER.
     84 173526  105737  177564          65$:    TSTB    @#TPS                   ; CHECK IF 'DONE FLAG' SET.
     85 173532  100003                          BPL     70$                     ; DO BRANCH IF 'DONE FLAG "NOT" SET'
     86
     87 173534  112437  177566                  MOVB    (R4)+,@#TPB             ; R4 = POINTS TO ADDRS OF DATA TO BE PRINTED
     88                                                                         ; AND COINTER IS THEN BUMPED.
     89 173540  000207                          RTS     PC                      ; RETURN TC MAIN LINE PROGRAM.
     90
     91 173542  005205                  70$:    INC     R5                      ; BUMP TIMER
     92 173544  001370                          BNE     65$                     ; IF TIMER IS STILL RUNNING DO BRANCH
     93
     94                                                                 ; **********************************************
     95 173546  000000                          HALT                            ; NO 'DONE FLAG' SET BY TERMINALIN TIME ALLOTED
     96                                                                 ; DEVICE ADDRESS = 177 564.
     97                                                                 ; **********************************************
     98 173550  005726                          TST     (SP)+                   ; POP STACK FOR ODT PROCEED.
     99 173552  000653                          BR      VTTEST                  ; PROCEED WILL RERUN VTTEST.
```

```
                                                        ; START OF CONSOL TERMINAL TABLES.
173554                          TABLE5:
173554  040                             .BYTE   040
173555  015                             .BYTE   CR
173556  012                             .BYTE   LF
173557  040                             .BYTE   040
173560  000                             .BYTE   0

173561                          TABLE6:
173561  062     070                     .ASCII  /28/                    ; 28K MEMORY
173563  062     064                     .ASCII  /24/                    ; 24K MEMORY
173565  062     060                     .ASCII  /20/                    ; 20K MEMORY
173567  061     066                     .ASCII  /16/                    ; 16K MEMORY
173571  061     062                     .ASCII  /12/                    ; 12K MEMORY
173573  070     040                     .ASCII  /8 /                    ; 8 K MEMORY
173575  064     040                     .ASCII  /4 /                    ; 4 K MEMORY
173577  077     077                     .ASCII  /??/                    ; UNEXPECTED MEMORY VALUE.
173601                          TABLE7:
173601  015                             .BYTE   CR                      ; DO A CARRAGE RETURN
173602  012                             .BYTE   LF                      ; DO A LINE FEED
173603  123     124     101             .ASCIZ  /START? /               ; TEXT DISPLAYED
173606  122     124     077
173611  040     000

173613  131                     TABL10: .BYTE   'Y                      ; IS IT A YES?
173614  116                             .BYTE   'N                      ; IS IT A NO??
                                .EVEN
```

CHANGE PAGE ROUTINE FOR PAGE SEVEN.

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE SEVEN.
                                ;* CHANGE PAGE ROUTINE FOR THIS PAGE.
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.

                                        ;FILL REMAIN ROM LOCATIONS WITH HALTS.
173616  000000                  .WORD   000000
173620  000000                  .WORD   000000
173622  000000                  .WORD   000000
173624  000000                  .WORD   000000
173626  000000                  .WORD   000000
173630  000000                  .WORD   000000
173632  000000                  .WORD   000000
173634  000000                  .WORD   000000
173636  000000                  .WORD   000000
173640  000000                  .WORD   000000
173642  000000                  .WORD   000000
173644  000000                  .WORD   000000
173646  000000                  .WORD   000000
173650  000000                  .WORD   000000
173652  000000                  .WORD   000000
173654  000000                  .WORD   000000
173656  000000                  .WORD   000000
173660  000000                  .WORD   000000
173662  000000                  .WORD   000000
173664  000000                  .WORD   000000
173666  000000                  .WORD   000000
173670  000000                  .WORD   000000
173672  000000                  .WORD   000000
173674  000000                  .WORD   000000
173676  000000                  .WORD   000000
173700  000000                  .WORD   000000
173702  000000                  .WORD   000000
173704  000000                  .WORD   000000
173706  000000                  .WORD   000000
173710  000000                  .WORD   000000
173712  000000                  .WORD   000000
173714  000000                  .WORD   000000
173716  000000                  .WORD   000000
173720  000000                  .WORD   000000
173722  000000                  .WORD   000000
173724  000000                  .WORD   000000
173726  000000                  .WORD   000000
173730  000000                  .WORD   000000
173732  000000                  .WORD   000000
```

CHANGE PAGE ROUTINE FOR PAGE SEVEN.

```
     59 173734  000000                  .WORD   000000
     60 173736  000000                  .WORD   000000
     61 173740  000000                  .WORD   000000
     62 173742  000000                  .WORD   000000
     63 173744  000000                  .WORD   000000
     64 173746  000000                  .WORD   000000
     65 173750  000000                  .WORD   000000
     66 173752  000000                  .WORD   000000
     67 173754  000000                  .WORD   000000
     68 173756  000000                  .WORD   000000
     69 173760  000000                  .WORD   000000
     70 173762  000000                  .WORD   000000
     71 173764  000000                  .WORD   000000
     72         173766                  .=173766
     74 173766  010437  177520          CPAGE7: MOV     R4,@#PAGELO             ; LOAD NEW PAGE WORD.
     75 173772  010507                          MOV     R5,PC                   ; JAM NEW PC.
     76
     77 173774  000000                          .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
     78
     79 173776  000000                          .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
     80                                                                         ; BY ROM BLASTING PROGRAM.
     81         000001'                 .END
```

TABLE OF CONTENTS

2- 1 BASIC DEFINITIONS
3- 1 LOCAL DATA
4- 1 SPECIAL EQUALITIES FOR BVD11-AA
5- 1 RLV-11 BOOTSTRAP
17- 1 CHANGE PAGE ROUTINE FOR PAGE ELEVEN.


```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS

        173000                  START   =       173000
        173514                  SWTEST  =       173514
        173550                  TABLE   =       173550
        173000                  ROBOOT  =       173000

000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
        173000                  . = 173000
```

```
                        .SBTTL  RLV-11 BOOTSTRAP
                        ;;*****************************************************************
                        ;*
                        ;* THE RLV-11 BOOTSTRAP WILL USE DRIVE 0 AS THE PRIMARY UNIT TO LOAD THE
                        ;* SECONDARY BOOTSTRAP. IF THE PRIMARY DRIVE IS NOT OPERATIONAL THEN
                        ;* A USER CAN LOAD THE LSI CPU REGISTER (R0) WITH THE DESIRED DRIVE
                        ;* NUMBER; 1, 2, 3 AND LOAD THE SECONDARY BOOTSTRAP.
                        ;*
                        ;* THE RLV BOOTSTRAP WILL PERFORM THE FOLLOWING FUNCTIONS:
                        ;*
                        ;* SET THE DEFAULT DRIVE VALUE,ZERO, IN R0.
                        ;* LOAD THE COMMAND AND STATUS REGISTER ADDRESS INTO R1 (174400).
                        ;* SET THE DIAGNOSTIC LIGHT DISPLAY TO "5". SET THE STACK TO 1100.
                        ;*
                        ;* RESET THE DRIVE. RESET THE CONTROLLER.
                        ;* READ THE HEADER FROM THE PRESENT TRACK.
                        ;* WAIT TWO MINUTES FOR THE DISK TO COME READY.
                        ;* CALCULATE THE DIFFERENCE TO TRACK ZERO AND DO SEEK.
                        ;*
                        ;* READ DATA FROM TRACK 0, SECTOR 0, HEAD 0, INTO MEMORY.
                        ;* LOAD MEMORY WITH DATA STARTING AT ADDRESS 000000.
                        ;* TEST LOCATION ZERO FOR A "NOP" INSTRUCTION AND IF CORRECT START.
                        ;*
                        ;* THE ROUTINE START WILL "BLANK" THE DIAGNOSTIC LIGHT DISPLAY.
                        ;* LOAD THE PAGE CONTROL REGISTER WITH PAGE 0.
                        ;* START THE SECONDARY BOOTSTRAP AT LOCATION ZERO.
                        ;*
                        ;* REFER TO THE ERROR ROUTINE IN THE LISTING FOA A DESCRIPTION OF ERROR HANDLING.
                        ;* ERROR REGISTER  VALUES
                        ;*
                        ;* R0= DRIVE UNIT SELECTED.
                        ;* R1= ADDRESS OF CSR.
                        ;* R2= ERROR RETRY COUNTER.
                        ;* R3= CONTENTS OF CSR.
                        ;* R4= COMMAND WORD.
                        ;* R5= ERROR STATUS REGISTER.
                        ;*
```

```
                        ;* CSR REGISTER VALUES FOR RLV.
                        ;* BIT 15       100 000 READ ONLY       ERROR
                        ;* BIT 14       040 000 READ ONLY       DRIVE ERROR.
                        ;* BIT 13       020 000 READ ONLY       NON-EXISTANT MEMORY.
                        ;* BIT 12       010 000 READ ONLY       DATA LATE OR HEADER NOT FOUND.
                        ;* BIT 11       004 000 READ ONLY       CRC ERROR ON HEADER OR DATA.
                        ;* BIT 10       002 000 READ ONLY       OPERIATION INCOMPLETE.
                        ;* BIT 9 - 8    001 400 READ/WRITE      DRIVE SELECT BITS.
                        ;* BIT 7        000 200 READ/WRITE      CONTROLLER READY.
                        ;* BIT 6        000 100 READ/WRITE      INTERRUPT ENABLE.
                        ;* BIT 5        000 040 READ/WRITE      BUS ADDRESS EXTENTION
                        ;* BIT 4        000 020 READ/WRITE      BUS ADDRESS EXTENTION.
                        ;* BIT 3        000 010 READ/WRITE      FUNCTION BITS.
                        ;* BIT 2        000 004 READ/WRITE      FUNCTION BITS.
                        ;* BIT 1        000 002 READ/WRITE      FUNCTION BITS.
                        ;* BIT 0        000 001 READ ONLY       DRIVE READY.
                        ;* FUNCTION BITS 1, 2, 3.
                        ;* FUNCTION     BIT SETTING             MODE
                        ;* 0            000 000                 MAINTENACNE MODE
                        ;* 1            000 002                 WRITE CHECK
                        ;* 2            000 004                 GET STATUS.
                        ;* 3            000 006                 SEEK.
                        ;* 4            000 010                 READ HEADER.
                        ;* 5            000 012                 WRITE DATA
                        ;* 6            000 014                 READ DATA.
                        ;* 7            000 016                 READ DATA WITHOUT HEADER CHECK.
                        ;* DRIVE ERROR STATUS REGISTER.
                        ;* BIT SETTING          ERROR
                        ;* 15                   WRITE DATA ERROR.
                        ;* 14                   HEADER CURRENT ERROR.
                        ;* 13                   WRITE LOCK.
                        ;* 12                   SEEK TIME OUT.
                        ;* 11                   SPIN ERROR.
                        ;* 10                   WRITE GATE ERROR.
                        ;* 9                    VOLUME CHECK.
                        ;* 8                    DRIVE SELECT ERROR.
                        ;* 7                    RESERVED (SET TO A  0  ).
                        ;* 6                    HEAD SELECT ERROR.
                        ;* 5                    COVER OPEN.
                        ;* 4                    HEADS OUT.
                        ;* 3                    BRUSH HOME.
                        ;* 0-2          ONE OF THE FOLLOWING ERRORS.
                        ;* 0                    LOAD STATE.
                        ;* 1                    SPIN -UP
                        ;* 2                    BRUSH CYCLE.
                        ;* 3                    LOAD HEADS.
                        ;* 4                    SEEK TRACK COUNTING.
                        ;* 5                    SEEK LINEAR MOTOR.
                        ;* 6                    UNLOAD HEADS.
                        ;* 7                    SPIN DOWN.
```

```
                                                ;ROM PAGE10, 004410 IN PAGE CONTROL REGISTER
                                                ;DIAGNOSTIC LIGHT DISPLAY=5
173000                          RLBOOT:
173000  005000                          CLR     R0              ;R0-DRIVE UNIT TO BOOT FROM.
173002  005010                          CLR     (R0)            ;MAKE LOACTION 0 =ZERO,FOR CORRECT READING OF
                                                                ;NOP INSTRUCTION OF SECONDARY BOOT.
173004  012701                          MOV     (PC)+,R1        ;R1=CSR ADDRESS
173006  174400                          .WORD   174400          ;STANDARD RLV I-O ADDRESS.
173010  012706  001100                  MOV     #1100,SP        ;SET STACK POINTER

173014  004767  000040          RLREDO: JSR     PC,RLINIT       ;SET DIAGNOSTIC LIGHT DISPLAY = 5
                                                                ;SET ERROR RETRY COUNTER = 10
                                                                ;SET CORRECT DRIVE SELECT BITS IN COMMAND WORD R4.

                                                ;RESET CONTROLLER AND DRIVE

173020  004767  000054          10$:    JSR     PC,RESDRV       ;RESET DRIVE.
173024  103003                          BCC     20$             ;BRANCH IF CARRY = 0, NO ERROR.
173026  077204                          SOB     R2,10$          ;CARRY = 1 ERROR, SUBTRACT 1 FROM ERROR RETRY
                                                                ;COUNTER AND RETRY UNTIL COUNTER IS ZERO.
                                        ;*************************************************************
                                        ;CANNOT CLEAR ERROR FLAGS AFTER CONTROLLER AND DRIVE RESET.
                                        ;RLERR1 ROUTINE WILL "HALT" OR JUMP TO EPROM ON ERROR.
173030  004767  000430                  JSR     PC,RLERR1       ;R0=DRIVE SELECTED.
                                                                ;R1=CSR ADDRESS.
                                                                ;R2=000000 ERROR RETRY COUNTER
                                                                ;R2=RLV BOOTSTRAP PAGE(IF EPROM SELECTED).
                                                                ;R3=CONTENTS OF CSR REGISTER
                                                                ;R4=COMMAND WORD
                                        ;*************************************************************

                                                ;READ HEADER

173034  012702  001100          20$:    MOV     #1100,R2        ;ERROR RETRY COUNTER WAIT FOR TWO
                                                                ;MINUTES OR DRIVE READY.
173040  004767  000060                  JSR     PC,RDHEAD       ;READ HEADER ON PRESENT CYLINDER
                                                                ;RETURN WITH HEADER INFORMATION IN R5.

                                                ;CALCULATE DIFFERENCE TO CYLINDER 0 AND SEEK

173044  004767  000130                  JSR     PC,SEEK         ;R5 MUST CONTAIN PRESENT HEADER INFO.

                                                ;READ DATA FROM CYL 0 HEAD 0 SECTOR 0

173050  004767  000174          30$:    JSR     PC,READ         ;READ DATA INTO MEMORY STARTING AT 00000C.

                                                ;TEST FOR PROPER BOOTSTRAP INDICATOR

                                                                ;"NOP" (000240) IN LOCATION IS STANDARD
173054  004767  000274                  JSR     PC,TST24        ;INDICATOR FOR A SECONDARY BOOTSTRAP.
                                                                ;IF NOP IS IN LOCATION ZERO THEN START IT.
```

```
                                                        ;INITIALIZE RLV BOOTSTRAP
173060                                  RLINIT:
173060  112737  000012  177524                MOVB    #FIVE,@#LIGHTS  ;DIAGNOSTIC LIGHT DISPLAY=5
173066  112702  000010                      MOVB    #10,R2          ;R2=ERROR RETRY COUNTER
173072  000300                            SWAB    R0              ;PLACE DRIVE NUMBER IN HIGH BYTE
173074  010004                            MOV     R0,R4           ;R4=COMMAND WORD FOR CSR
173076  000207                            RTS     PC              ;BITS 8 AND 9 ARE DRIVE SELECT.
                                                                  ;END OF RLINIT ROUTINE

                                                        ;RE-SET DRIVE AND THEN READ STATUS
173100                                  RESDRV:
173100  042704                            BIC     (PC)+,R4        ;CLEAR FUNCTION BITS
173102  000016                            .WORD   BIT1!BIT2!BIT3

173104  052704                            BIS     (PC)+,R4        ;RE-SET DRIVE WILL CLEAR DRIVE ERRORS AND READ
173106  000004                            .WORD   BIT2            ;DRIVE STATUS.SET FUNCTION BITS IN COMMAND WORD.
173110  012761  000013  000004            MOV     #13,4(R1)       ;A DRIVE RESET IN A GET STATUS WITH (DA) SET.
                                                                  ;SET DISK ADDRESS (DA) TO 13.
173116  004767  000274                    JSR     PC,RLGO         ;"GO" WILL DO FUNCTION IN R4
                                                                  ;IF NO RETURN IS MADE, THEN A ERROR IN THE
                                                                  ;RLGO ROUTINE.
                                                                  ;REFER TO THE RLGO ROUTINE FOR CAUSE OF ERROR.
                                                                  ;REGISTERS VALUES WILL BE
                                                                  ;CARRY BIT = 1 ERROR
                                                                  ;CARRY BIT = 0 NO ERROR
                                                                  ;R0 = DRIVE UNIT
                                                                  ;R1 = CSR ADDRESS
                                                                  ;R2 = ADDRESS OF ERROR CALL.
                                                                  ;R3 = CSR CONTENTS
                                                                  ;R4 = CSR ADDRESS
173122  000207                            RTS     PC              ;RETURN TO CALLER WITH INDICATORS SET BY RLGO.
```

```
                                                        ;READ HEADER ROUTINE
173124                          RDHEAD:
173124  042704                        BIC     (PC)+,R4        ;CLEAR FUNCTION BITS
173126  000016                        .WORD   BIT1!BIT2!BIT3
                
173130  052704                        BIS     (PC)+,R4
173132  000010                        .WORD   BIT3            ;SET COMMAND FUNCTION TO READ A HEADER.
                
173134  004767  000256                JSR     PC,RLGO         ;DO COMMAND IN R4 RETURN WITH THE FOLLOWING
173140  103014                        BCC     40$             ;CARRY = 0 NO ERROR. DO BRANCH.
                                                            ;CARRY = 1 ERROR.
                                                            ;TEST THAT ERROR IS LEGAL ONE.
173142  032703  040000                BIT     #BIT14,R3       ;TEST IF DRIVE ERROR.
173146  001002                        BNE     20$             ;DRIVE ERROR IS BIT 14,BRANCH IF DRIVE ERROR.
                
                                                        ;CONTROLLER ERROR
173150  010305                        MOV     R3,R5           ;R5 = CONTENTS OF CSR.
173152  000402                        BR      35$             ;GO TO ERROR RETRY  ROUTINE
                
                                                        ;DRIVE ERROR
                
173154  004767  000140          20$:  JSR     PC,RDSTAT       ;GET DRIVE STATUS
                                                            ;THIS ROUTINE WILL LOAD R5 WITH DRIVE STATUS
                
173160  004767  177714          35$:  JSR     PC,RESDRV       ;THERE IS A ERROR. RESET DRIVE
                
173164  077221                        SOB     R2,RDHEAD       ;CARRY = 1 ERROR
                                                            ;SUBTRACT 1 FROM RETRY COUNTER AND IF
                                                            ;NOT ZERO REDO READ HEADER COMMAND.
                                                    ;READ HEADER ERROR
                                                    ;COULD NOT READ A HEADER.ERROR RETRY COUNTER IS ZERO.
173166  004767  000272                JSR     PC,RLERR1       ;A LOAD DEVICE ERROR HAS OCCURED,JUMP TO
                                                            ;ERROR ROUTINE. TEST FOR FUTHER PROGRAM
                                                            ;DIALOGUE OR EXECUTE A "HALT".
                                                            ;IF "HALT" IS EXECUTED AND THE USER "PROCEEDS"
                                                            ;THE PROGRAM WILL RERUN THE BOOTSTRAP.
                                                            ;IF THE "EPROM" IS SELECTED REFER TO
                                                            ;BOOTSTRAP ERROR ROUTINE FOR MORE INFORMATION
                                                            ;ON "EPROM" EXECUTION.
                                                            ;ERROR REGISTER CONTAIN
                                                            ;R0 = DRIVE UNIT
                                                            ;R1 = CSR ADDRESS
                                                            ;R2 = ERROR RETRY COUNTER
                                                            ;R3 = CONTENTS OF CSR
                                                            ;R4 = COMMAND WORD
                                                            ;R5 = DRIVE STATUS
                                                            ;R5 = CONTENTS OF CSR IF CONTROLLER ERROR.
                
                                                            ;MOVE HEADER DATA TO R5.
173172  016105  000006          40$:  MOV     6(R1),R5        ;R5 CONTAINS HEADER WORD ON PRESENT CYLINDER.
173176  000207                        RTS     PC              ;RETURN TO CALLER WITH INDICATORS SET BY RLGO.
```

```
                                                        ;SEEK ROUTINE
                                                        ;R5 MUST CONTAIN PRESENT HEADER INFO
                                                        ;TO PERFORM A SEEK DIFFERENCE MUST BE
                                                        ;LOADED INTO DISK ADDRESS (RLDA) WITH
                                                        ;BIT 0 SET BIT 1 AND 15 CLEAR
                                                        ;CYLINDER DIFFERENCE IN BITS 7 - 14
173200                          SEEK:
173200  012702  000010                MOV     #10,R2          ;RESET ERROR RETRY COUNTER
173204                          10$:
173204  042704                        BIC     (PC)+,R4        ;CLEAR FUNCTION BITS
173206  000016                        .WORD   BIT1!BIT2!BIT3

173210  052704                        BIS     (PC)+,R4        ;MAKE COMMAND WORD FUNCTION BITS= 3
173212  000006                        .WORD   BIT1!BIT2       ;THIS IS FUNCTION FOR SEEK.

173214  042705                        BIC     (PC)+,R5        ;CLEAR THE FOLLOWING BITS IN THE HEADER WORD.
173216  100177                        .WORD BIT15!BIT6!BIT5!BIT4!BIT3!BIT2!BIT1!BIT0

                                                        ;THIS WILL LEAVE ONLY THE CYLINDER ADDRESS.

173220  005205                        INC     R5              ;BIT0 MUST BE SET IN (DA) TO DO A SEEK.
173222  010561  000004                MOV     R5,4(R1)        ;SET CYLINDER DIFFERENCE IN DISK ADDRESS

173226  004767  000164                JSR     PC,RLGO         ;DO FUNCTION. RETURN WITH FOLLOWING
173232  103005                        BCC     20$             ;CARRY = 0 NO ERRORS DO BRANCH
                                                        ;CARRY = 1 RETRY 10 TIMES
173234  004767  177664                JSR     PC,RDHEAD       ;FIRST READ HEADER
173240  077217                        SOB     R2,10$          ;SUBTRACT 1 FROM ERROR RETRY COUNTER
                                                        ;AND IF NOT ZERO RETRY SEEK.
                                                ;*****************************************************************
                                                ;SEEK ERROR

173242  004767  000216                JSR     PC,RLERR1       ;A LOAD DEVICE ERROR HAS OCCURED,JUMP TO
                                                        ;ERROR ROUTINE. TEST FOR FUTHER PROGRAM
                                                        ;DIALOGUE OR EXECUTE A "HALT".
                                                        ;IF "HALT" IS EXECUTED AND THE USER "PROCEEDS"
                                                        ;THE PROGRAM WILL RERUN THE BOOTSTRAP.
                                                        ;IF THE "EPROM" IS SELECTED REFER TO
                                                        ;BOOTSTRAP ERROR ROUTINE FOR MORE INFORMATION
                                                        ;ON "EPROM" EXECUTION.
                                                        ;ERROR REGISTER CONTAIN
                                                        ;R0 = DRIVE UNIT
                                                        ;R1 = CSR ADDRESS
                                                        ;R2 = ERROR RETRY COUNTER
                                                        ;R3 = CONTENTS OF CSR
                                                        ;R4 = COMMAND WORD
                                                        ;R5 = DRIVE STATUS IF DRIVE ERROR.
                                                        ;R5 = CONTENTS OF CSR WITH CONTROLLER ERROR.
                                                ;*****************************************************************
                                                ;CLEAR FUNCTION BITS IN COMMAND WORD.
173246  000207                20$:    RTS     PC              ;RETURN TO CALLER WITH INDICATORS SET BY RLGO.
```

```
                                                        ;READ CYL 0 SECTOR 0 HEAD 0

 173250                                 READ:
 173250  012702  000010                        MOV     #10,R2          ;RESET ERROR RETRY COUNTER
                                                                        ;TO DO A READ THE FOLLOWING MUST BE SET-UP
 173254                                 10$:
 173254  042704                                BIC     (PC)+,R4        ;CLEAR FUNCTION BITS
 173256  000016                                .WORD   BIT1!BIT2!BIT3

 173260  052704                                BIS     (PC)+,R4        ;MAKE COMMAND WORD FUNCTION BITS=6.
 173262  000014                                .WORD   BIT2!BIT3       ;FUNCTION IS READ DATA.

 173264  005061  000002                        CLR     2(R1)           ;BASE ADDRESS = 000 000

 173270  005061  000004                        CLR     4(R1)           ;DISK ADDRESS = SECTOR 0 - HEAD 0 - CYL 0 -

 173274  012761  177400  000006                MOV     #177400,6(R1)   ;(MP)WORD COUNT = 400 ( 177400 TWO'S COMPLEMENT VALUE NEEDED

 173302  004767  000110                        JSR     PC,RLGO         ;DO READ. GO WILL RETURN WITH THE FOLLOWING.
 173306  103003                                BCC     20$             ;CARRY = 0 NO ERROR
 173310  077217                                SOB     R2,10$          ;CARRY = 1 ERROR RETRY 10 TIMES

                                                        ;*****************************************************************
                                                        ;ERROR COULD NOT READ CYL 0 SECTOR 0 HEAD 0
 173312  004767  000146                        JSR     PC,RLERR1       ;A LOAD DEVICE ERROR HAS OCCURED,JUMP TO
                                                                        ;ERROR ROUTINE. TEST FOR FUTHER PROGRAM
                                                                        ;DIALOGUE OR EXECUTE A "HALT".
                                                                        ;IF "HALT" IS EXECUTED AND THE USER "PROCEEDS"
                                                                        ;THE PROGRAM WILL RERUN THE BOOTSTRAP.
                                                                        ;IF THE "EPROM" IS SELECTED REFER TO
                                                                        ;BOOTSTRAP ERROR ROUTINE FOR MORE INFORMATION
                                                                        ;ON "EPROM" EXECUTION.
                                                                        ;ERROR REGISTER CONTAIN:
                                                                        ;R0 = DRIVE UNIT
                                                                        ;R1 = CSR ADDRESS
                                                                        ;R2 = ERROR RETRY COUNTER
                                                                        ;R3 = CONTENTS OF CSR
                                                                        ;R4 = COMMAND WORD
                                                                        ;R5 = DRIVE STATUS IF DRIVE ERROR.
                                                                        ;R5 = CONTENTS OF CSR IF CONTROLLER ERROR.
                                                        ;*****************************************************************

                                                        ;ROUTINE TO CLEAR THE FUNCTION BITS IN THE

 173316  000207                         20$:   RTS     PC              ;EXIT ROUTINE FOR READ.
```

```
                                                        ;READ ERROR STATUS
                                        RDSTAT:
173320                                          MOV     R4,R5           ;DO NOT DESTROY ORIGINAL COMMAND WORD

173322  042705                                  BIC     (PC)+,R5        ;USE R5 TO TEST FOR ERROR IN DRIVE.
173324  000016                                  .WORD   BIT1!BIT2!BIT3  ;CLEAR FUNCTION BITS

173326  052705                                  BIS     (PC)+,R5        ;SET FUNCTION BITS TO DO GET STATUS
173330  000004                                  .WORD   BIT2

173332  012761  000003  000004                  MOV     #3,4(R1)        ;SET DISK ADRESS REGISTER(DA) TO 3 READ STATUS.

173340  010511                                  MOV     R5,(R1)         ;EXECUTE COMMAND IN R5
173342  004767  000052                          JSR     PC,RLGO1        ;CHECK FOR CONTROLLER READY.

173346  016105  000006                          MOV     6(R1),R5        ;MULTI-PURPOSR REGISTER CONTAINS
173352  000207                                  RTS     PC              ;EXIT READ STATUS
```

```
                                                ;TEST FOR SECONDARY BOOTSTRAP.
173354                          TST24:
173354  112737  000011  177524          MOVB    #SIX,@#LIGHTS           ;DIAGNOSTIC LIGHT DISPLAY = 6
173362  022737  000240  000000          CMP     #240,@#0        ;TEST LOCATION ZERO FOR A "NOP" (240)
173370  001402                          BEQ     10$             ;BRANCH IF NOP THERE

                                                ;*****************************************************************
                                                ;SECONDARY BOOTSTRAP REQUIRES A NOP IN LOCATION 0.  NONE THERE.

173372  004767  000066                  JSR     PC,RLERR1       ;A LOAD DEVICE ERROR HAS OCCURED,JUMP TO
                                                ;ERROR ROUTINE. TEST FOR FUTHER PROGRAM
                                                ;DIALOGUE OR EXECUTE A "HALT".
                                                ;IF "HALT" IS EXECUTED AND THE USER "PROCEEDS"
                                                ;THE PROGRAM WILL RERUN THE BOOTSTRAP.
                                                ;IF THE "EPROM" IS SELECTED REFER TO
                                                ;BOOTSTRAP ERROR ROUTINE FOR MORE INFORMATION
                                                ;ON "EPROM" EXECUTION.
                                                ;ERROR REGISTER CONTAIN:
                                                ;R0 = DRIVE UNIT
                                                ;R1 = CSR ADDRESS
                                                ;R2 = ERROR RETRY COUNTER
                                                ;R3 = CONTENTS OF CSR
                                                ;R4 = COMMAND WORD
                                                ;R5 = DRIVE STATUS
                                                ;*****************************************************************

173376  112737  000037  177524  10$:    MOVB    #BLANK,@#LIGHTS ;CLEAR DIAGNOSTIC LIGHT DISPLAY
173404  005005                          CLR     R5              ;START AT LOCATION 0
173406  012704  000400                  MOV     #PAGE0,R4       ;PAGE RE-SET TO ZERO
173412  000167  000350                  JMP     CPAG11          ;BYE
```

```
                                                        ;GO ROUTINE
                                                                ;THIS ROUTINE WILL EXECUTE THE COMMAND WORD.
                                                                ;CHECK FOR CONTROLLER READY
                                                                ;CHECK FOR ERROR BIT SET AND SET PROGRAM ERROR FLAG
                                                                ;LOAD CSR CONTENTS INTO R3.

173416                                  RLGO:
173416  010411                          MOV     R4,(R1)         ;EXECUTE COMMAND IN R4

173420  005003                  RLGO1:  CLR     R3              ;CALL TO HERE WHEN ANOTHER ROUTINE LOADED CSR
173422  012746  000100                  MOV     #100,-(SP)      ;CLEAR WATCHDOG TIMER AND SET EXTENDED TIMER .
173426  105711                  10$:    TSTB    (R1)            ;TEST FOR CONTROLLER DONE FLAG.
173430  100407                          BMI     20$             ;BRANCH WHEN SET
173432  005203                          INC     R3              ;FLAG NOT SET BUMP TIMER AND TEST FLAG AGAIN.
173434  001374                          BNE     10$             ;BRANCH AS LONG AS TIME STILL REMAINS
173436  005316                          DEC     (SP)            ;SUBTRACT ONE FROM EXTENDED TIMER AND BRANCH
173440  001372                          BNE     10$             ;BACK AS LONG AS EXTENDED COUNTER DOES NOT GET TO ZERO
                                                        ;*******************************************************
                                                                ;TIME OUT ERROR
173442  011103                          MOV     (R1),R3         ;NO CONTROLLER DONE FLAG SET IN TIME ALLOWED.
173444  004767  000014                  JSR     PC,RLERR1       ;TIME OUT ERROR.
                                                                ;NO CONTROLLER READY IN TIME ALLOWED.
                                                                ;ERROR REGISTERS ARE
                                                                ;R0= DRIVE SELECTED
                                                                ;R1= ADDRESS OF CSR
                                                                ;R2= ADDRESS OF ERROR CALL.
                                                                ;R3= CONTENTS OF CSR
                                                                ;R4= COMMAND WORD
                                                        ;*******************************************************

173450                          20$:
173450  005726                          TST     (SP)+           ;CORRECT STACK FOR PROPER RETURN.
173452  005711                          TST     (R1)            ;CONTROLLER ERROR FLAG SET
173454  100001                          BPL     30$             ;TEST FOR ERROR BIT15 . BRANCH IF NOT SET.
173456  000261                          SEC                     ;ERROR FLAG SET. SET CARRY = 1.
173460  011103                  30$:    MOV     (R1),R3         ;LOAD R3 WITH CONTENTS OF CSR
173462  000207                          RTS     PC              ;EXIT ROUTINE
                                                                ;CARRY = 0 NO ERROR
                                                                ;CARRY = 1 ERROR
                                                                ;R0= DRIVE SELECTED
                                                                ;R1= ADDRESS OF CSR
                                                                ;R2= ERROR RETRY COUNTER
                                                                ;R3= CONTENTS OF CSR
                                                                ;R4= COMMAND WORD
```

```
                                                        ;RLV1 LOAD DEVICE ERROR ROUTINE.
                                                        ;TEST IF PROGRAM IS TO "HALT" OR JUMP TO "EPROM".
173464                                  RLERR1:
173464  032737  003400  177524          BIT     #BIT8!BIT9!BIT10,@#SWITCH
                                                        ;TEST FOR THE "ROM BOOT" .
173472  001006                          BNE     20$     ;BRANCH IF ROM IS SELECTED OTHERWISE "HALT"
                                                        ;BIT8= RESERVED DIAGNOSTIC ROM
                                                        ;BIT9= EPROM SELECTED
                                                        ;BIT10= SYSTEM ROM

                                                ;*****************************************************************

173474  004767  177620          10$:    JSR     PC,RDSTAT       ;PUT DRIVE STATUS REGISTER IN R5.
173500  011602                          MOV     (SP),R2         ;MAKE R2 ADDRESS OF ERROR CALL

173502  000000                          HALT            ;RLV  DEVICE ERROR.


                                                        ;THE ERROR WAS CAUSED BY A RLV LOAD DEVICE.
                                                        ;STACK POINTS TO ROUTINE THAT CALLED ERROR "HALT".
173504  000167  177304                  JMP     RLREDO          ;PROCEEDING FROM THIS "HALT" WILL RERUN THE
                                                        ;RLV BOOTSTRAP. REFER TO SPECIFIC ERROR
                                                        ;FOR CAUSE OF FAILURE.

                                                        ; R0 = DEVICE SELECTED
                                                        ; R1 = CSR ADDRESS
                                                        ; R2 = ADDRESS OF ERROR CALL.
                                                        ; R3 = CONTENTS OF CSR
                                                        ; R4 =  COMMAND WORD.
                                                        ; R5 = ERROR STATUS REGISTER OF DRIVE
                                                ;*****************************************************************
173510                          20$:
173510  012704  007416                  MOV     #PAGE16,R4      ; PAGE CONTROL REGISTER FOR ROM BOOT
173514  012705  173000                  MOV     #ROBOOT,R5      ; PUT ROM BOOTSTRAP ADDRESS
                                                        ; IN R5
173520  000522                          BR      CPAG11          ; BRANCH TO CHANGE PAGE ROUTINE
```

```
                                        ;MAINTENANCE MODE ONLY
173522                          HELPRL:
173522  005000                  CLR     R0              ;SET DEFAULT DRIVE NUMBER TO 0.
173524  012701                  MOV     (PC)+,R1        ;SET UP CORRECT CSR ADDRESS
173526  174400                  .WORD   174400
173530  012702  000010          MOV     #10,R2          ;SET ERROR RETRY COUNTER
173534  012706  004000          MOV     #4000,SP        ;SET STACK POINTER OUT OF DATA AREA.
173540  004767  177334          JSR     PC,RESDRV       ;RESET THE DRIVE

173544  103001                  BCC     20$             ;BRANCH IF NO ERRORS.

173546  000000                  HALT                    ;ERROR IN RESETTING ERROR BITS IN CSR.
                                ;WRITE TO DISK CYL 0 SECTOR 0 HEAD 0
173550  005005          20$:    CLR     R5

173552  000000                  HALT                    ;LOAD DATA PATTERN TO BE WRITTEN IN R5

173554  012703  000400          MOV     #400,R3         ;WRITE PATTERN IN MEMORY LOCATION 1000-1776
173560  012704  002000          MOV     #2000,R4
173564  010544          10$:    MOV     R5,-(R4)
173566  005303                  DEC     R3
173570  001375                  BNE     10$
173572  005105                  COM     R5              ;WRITE COMPLEMENT OF DATA PATTERN
173574  010544          30$:    MOV     R5,-(R4)        ;INTO MEMORY LOACTIONS 000-376
173576  005704                  TST     R4              ;TEST FOR LOCATION 0
173600  001375                  BNE     30$             ;BRANCH IF NOT ZERO
                                ; RESET COMMAND WORD
173602  004767  177252          JSR     PC,RLINIT       ;RESET COMMAND WORD IN R4.
173606  004767  177312          JSR     PC,RDHEAD       ;READ HEADER
173612  004767  177362          JSR     PC,SEEK         ;SEEK TO CYLINDER ZERO.
                                ;WRITE REQUIRES THE FOLLOWING SET-UP
173616  012761  001000  000002  MOV     #1000,2(R1)     ;BASE ADDRESS = 001 000
173624  005061  000004          CLR     4(R1)           ;DISK ADDRESS = 000 000
173630  012761  177400  000006  MOV     #177400,6(R1)   ;(MP) WORD COUNT = 400
173636  142704                  BICB    (PC)+,R4        ;CLEAR FUNCTION BITS
173640  000016                  .WORD   BIT1!BIT2!BIT3
173642  152704                  BISB    (PC)+,R4        ;COMMAND TO WRITE IN R4
173644  000012                  .WORD   000012

173646  004767  177544          JSR     PC,RLGO         ;GO WRITE.  RETURN WITH THE FOLLOWING
173652  103002                  BCC     40$             ;CARRY = 0 NO ERROR
                                                        ;CARRY = 1 ERROR
                                ;*************************************************************
173654  000000                  HALT                    ;WRITE ERROR
```

```
     58 173656  000721                        BR      HELPRL          ;PROCEED WILL RE RUN WRITE ROUTINE
     59                                                                ;R0 = DRIVE UNIT
     60                                                                ;R1 = CSR ADDRESS
     61                                                                ;R2 = ERROR RETRY COUNTER
     62                                                                ;R3 = CONTENTS OF CSR
     63                                                                ;R4 = COMMAND WORD
     64                                                                ;R5 = DRIVE STATUS
     65                                                              ;*****************************************************************
     66 173660  000167  177130          40$:  JMP     RLREDO          ;EXIT
     67                                                                ;USE BOOTSTRAP ROUTINE TO READ DATA
     68                                                                ;WRITTEN AND WILL HALT WHEN TESTING FOR
     69                                                                ;THE NOP INSTRUCTION.
```

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE ELEVEN.
                                ;* CHANGE PAGE ROUTINE FOR THIS PAGE.
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.

                                                ;FILL REMAINING ROM LOCATIONS WITH HALTS.

 173664  000000                .WORD   000000
 173666  000000                .WORD   000000
 173670  000000                .WORD   000000
 173672  000000                .WORD   000000
 173674  000000                .WORD   000000
 173676  000000                .WORD   000000
 173700  000000                .WORD   000000
 173702  000000                .WORD   000000
 173704  000000                .WORD   000000
 173706  000000                .WORD   000000
 173710  000000                .WORD   000000
 173712  000000                .WORD   000000
 173714  000000                .WORD   000000
 173716  000000                .WORD   000000
 173720  000000                .WORD   000000
 173722  000000                .WORD   000000
 173724  000000                .WORD   000000
 173726  000000                .WORD   000000
 173730  000000                .WORD   000000
 173732  000000                .WORD   000000
 173734  000000                .WORD   000000
 173736  000000                .WORD   000000
 173740  000000                .WORD   000000
 173742  000000                .WORD   000000
 173744  000000                .WORD   000000
 173746  000000                .WORD   000000
 173750  000000                .WORD   000000
 173752  000000                .WORD   000000
 173754  000000                .WORD   000000
 173756  000000                .WORD   000000
 173760  000000                .WORD   000000
 173762  000000                .WORD   000000
 173764  000000                .WORD   000000
         173766                .=173766
 173766  010437  177520  CPAG11: MOV     R4,@#PAGELO             ; LOAD NEW PAGE WORD.
 173772  010507                 MOV     R5,PC                   ; JAM NEW PC.

 173774  000000                 .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
```

CHANGE PAGE ROUTINE FOR PAGE ELEVEN.

```
      60
      61 173776  000000                          .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
      62                                                                         ; BY ROM BLASTING PROGRAM.
      63          000001                  .END
```

TABLE OF CONTENTS

  2- 1 BASIC DEFINITIONS
  3- 1 LOCAL DATA
  4- 1 SPECIAL EQUALITIES FOR BVD11-AA
  5- 1 FLOPPY BOOT
 17- 1 CHANGE PAGE ROUTINE FOR PAGE THIRTEEN.


```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS

173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550
173000                          ROBOOT  =       173000

000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          .=173000
```

```
                        .SBTTL  FLOPPY BOOT
                        ; ************************************************************************************
                        ; ************************************************************************************
                        ;* FLOPPY BOOT
                        ;* THE BOOTSTRAP FOR THE FLOPPY RX01 WILL CONTAIN THE FOLLOWING:
                        ;* ISSUE A RESET CPU COMMAND, THIS WILL INITIALIZE THE FLOPPY.
                        ;* CHECK FOR "DONE" AFTER RESET ISSUED.
                        ;* HAVE A TIME ENABLED FOR NO FLOPPY "DONE"
                        ;* AFTER "DONE" CHECK IF DRIVE 0 READY IF NOT ASSUME   DRIVE 1 READY.
                        ;* SELECT PROPER UNIT NUMBER AND STORE IT IN R0
                        ;* STORE COMMAND/STATUS WORD IN R1
                        ;* SET UP SECTOR 1
                        ;* SET UP TRACK 1
                        ;* READ DATA FROM DRIVE CHECK FOR ERRORS.
                        ;* IF DRIVE 1 NOT READY THIS IS WHERE IT WILL BE FOUND.
                        ;* SET WATCH DOG TIMER FOR "DONE" ON FIRST FUNCTION SENT TO DRIVE.
                        ;* LOAD MEMORY WITH 1 SECTOR WORTH OF DATA STARTING AT LOCATION 000 000.
                        ;* CHECK FOR A NOP[000 240] IN LOCATION 000 000.
                        ;* IF NO STATUS ERRORS AND NOP CORRECT START BOOTSTRAP AT 000 000.
                        ;* IF STATUS ERROR DISPLAY A 10  IN THE LED'S
                        ;* IF NOP NOT IN LOCATION 000 000 DISPLAY A 14 IN LED'S
                        ;* REGISTER  VALUES
                        ;* INPUT
                        ;* R0= TEST BIT MATCH FOR THIS ROUTINE.
                        ;* R1= NEXT ENTRY INTO SWITCH CHECKING TABLE.
                        ;* R2= RESULT OF SOME TEST.
                        ;* R3= RESULT OF SOME TEST.
                        ;* R4= ADDRESS OF FLOPPY BOOT.
                        ;* R5= PAGE CONTROL REGISTER.
                        ;* ON OUTPUT TO FLOPPY BOOT
                        ;* R0= UNIT LOADED FROM.
                        ;* R1= COMMAND / STATUS REGISTER ADDRESS
                        ;* R2= NOT USED.
                        ;* R3= WATCHDOG TIMER
                        ;* R4= COMMAND REGISTER SETTINGS
                        ;* R5= ADDRESS FOR STORING DATA IN MEMORY.
                        ;* ERROR REGISTER INFORMATION
                        ;* R0= ADDRESS OF ERROR CALL.
                        ;* R1= RXCS ADDRESS.
                        ;* R2= UNIT SELECTED.
                        ;* R3= DEFINITIVED ERROR CODES.
                        ;* R4= ERROR STATUS REGISTER.
                        ;* FLOPPY CSR REGISTER VALUES.
                        ;* BIT 15       100 000          READ ONLY       ERROR
                        ;* BIT 14       040 000          WRITE ONLY      INITIALIZE FLOPPY DONE.
                        ;* BIT 13       020 000          WRITE ONLY      EXTENDED ADDRESS BITS RX02.
                        ;* BIT 12       010 000          WRITE ONLY      EXTENDED ADDRESS BITS RX02.
                        ;* BIT 11       004 000          READ  ONLY      INDICATOR FOR A RX02 SYSTEM.
                        ;* BIT 10       002 000          WRITE ONLY      RESERVED = 0
                        ;* BIT 9        001 000          WRITE ONLY      RESERVED = 0.
                        ;* BIT 8        C00 400          READ/WRITE      DENSITY RX02
```

```
                        ;* FLOPPY CSR REGISTER VALUES
                        ;*
                        ;* BIT 7        000 200        READ  ONLY      TRANSFER REQUEST.
                        ;* BIT 6        000 100        READ/WRITE      INTERRUPT ENABLE.
                        ;* BIT 5        000 040        READ  ONLY      DONE
                        ;* BIT 4        000 020        READ/WRITE      UNIT SELECTED.
                        ;* BIT 3        000 010        WRITE ONLY      FUNCTION BITS
                        ;* BIT 2        000 004        WRITE ONLY      FUNCTION BITS
                        ;* BIT 1        000 002        WRITE ONLY      FUNCTION BITS
                        ;* BIT 0        000 001        WRITE ONLY      GO
                        ;*
                        ;* FUNCTION BIT
                        ;*
                        ;* FUNCTION     BIT SETTING     MODE
                        ;*
                        ;*      0       000 000         FILL
                        ;*      1       000 002         EMPTY
                        ;*      2       000 004         WRITE SECTOR.
                        ;*      3       000 006         READ  SECTOR.
                        ;*      4       000 010         SET MEDIA DENSITY.(RX02)
                        ;*      5       000 012         MAINTENANCE READ STATUS.(250 MSEC.)
                        ;*      6       000 014         WRITE SECTOR WITH DELETED DATA.
                        ;*      7       000 016         READ ERROR CODE.
                        ;*
                        ;* ERROR AND STATUS REGISTER
                        ;*
                        ;* BIT SETTING                  MODE
                        ;*      11      004 000         NONEXISTANT MEMORY.(RX02)
                        ;*      10      002 000         WORD COUNT OVERFLOW.(RX02)
                        ;*      9       001 000
                        ;*      8       000 400
                        ;*
                        ;*      7       000 200         DRIVE READY
                        ;*      6       000 100         DELETED DATA
                        ;*      5       000 040
                        ;*      4       000 020         DENSITY ERROR (RX02)
                        ;*      3       000 010         RX02 AC LOW.
                        ;*      2       000 004         INITIALIZE DONE.
                        ;*      1       000 002         PARITY ERROR RX01.
                        ;*      0       000 001         CRC ERROR.
                        ;*
                        ;*
                        ; **********************************************************************************
                        ; **********************************************************************************
```

```
                                                ;ROM PAGE IS PAGE12.  005412 IN PAGE CONTROL REGISTER
                                                        ;DIAGNOSTIC LIGHT DISPLAY = 5
173000                          RXBOOT:
173000  005000                          CLR     R0              ;R0= DRIVE USED TO LOAD SECONDARY BOOT
173002  005010                          CLR     (R0)            ;CLEAR DATA IN 0.
173004  012701                          MOV     (PC)+,R1        ;R1= ADDRESS OF CSR
173006  177170                          .WORD   177170          ;
173010  012706  001100                  MOV     #1100,SP        ;BEGINNING OF BOOTSTRAP

173014  004767  000062          RXREDO: JSR     PC,CONFIG       ;TURN ON DIAGNOSTIC LIGHTS
                                                                ;SET STACK POINTER TO 1100
                                                                ;R4 = CONFIGURE COMMAND WORD.
                                                ;READ STATUS FOR DRIVE READY.
173020  012702  000740                  MOV     #740,R2         ;MAKE WAIT LOOP 1 MINUTE 30 SECONDS
173024  004767  000354          10$:    JSR     PC,RXSTAT       ;READ STATUS TAKES 250 MS.
173030  000241                          CLC                     ;CLEAR CARRY BIT.
173032  032761  000200  000002          BIT     #BIT7,2(R1)     ;TEST BIT 7 OF STATUS REGISTER(DRIVE READY)
173040  001003                          BNE     15$             ;BRANCH IF DRIVE IS READY
173042  077210                          SOB     R2,10$          ;BRANCH BACK AND READ STATUS AGAIN LOOK FOR READY.
173044  004767  000432                  JSR     PC,RXERR        ;ERROR NO DRIVE READY.
                                                                ;PROCEEDING FROM HALT WILL RERUN RXBOOT.
                                                                ;ERROR REGISTER VALUES ARE:
                                                                ;R0 = DRIVE SELECTED
                                                                ;R1 = ADDRESS OF CSR
                                                                ;R2 = ERROR RETRY COUNTER
                                                                ;R3 = CONTENTS OF CSR
                                                                ;R4 = COMMAND WORD
                                                                ;R5 = ERROR STATUS REGISTER
                                                ;READ
173050  052704                  15$:    BIS     (PC)+,R4        ;FUNCTION BITS FOR A "READ"
173052  000006                          .WORD   BIT1!BIT2
173054  004767  000062                  JSR     PC,RXRDWR       ;GO DO READ
                                                                ;ROUTINE WILL "HALT" AFTER 10 ATTEMPTS
                                                                ;TO DO READ PROCEED WILL RERUN RXBOOT.
                                                ;EMPTY COMMAND
173060  052704                  20$:    BIS     (PC)+,R4        ;SETUP BEFORE CALL IS MADE
173062  000002                          .WORD   BIT1
173064  005005                          CLR     R5              ;R2=ERROR RETRY COUNTER
173066  004767  000120                  JSR     PC,FILEPT       ;R4= FUNCTION BIT FOR EMPTY
173072  103772                          BCS     20$             ;R5 = BUS ADDRESS
                                                                ;FILEPT ROUTINE WILL RETURN WITH
                                                                ;A ERROR FLAG (CARRY=1). IF ERROR
                                                                ;FLAG SET SETUP ALL BUT ERROR COUNTER
                                                                ;AND RETRY EMPTY COMMAND
                                                                ;AFTER 10 TRYS AND ERROR IS STILL
                                                                ;PRESENT "HALT" IN "FILEPT" ROUTINE
                                                                ;PROCEEDING FROM HALT WILL RERUN
                                                                ;RX BOOT STRAP.

                                                ;TEST THAT A SECONDARY BOOTSTRAP WAS LOADED
173074  004767  000340                  JSR     PC,TST240       ;THIS ROUTINE WILL CHANGE THE
```

```
     58 173100  000745                        BR      RXREDO          ;DIAGNOSTIC LIGHT DISPLAY TO 6
     59                                                                 ;AND TEST LOCATION ZERO. IT
     60                                                                 ;SHOULD HAVE A "NOP" (000240)
     61                                                                 ;WHICH IS A STANDARD REQUIREMENT
     62                                                                 ;FOR BOOTSRTAPS.  IF A "NOP" IS
     63                                                                 ;NOT PRESENT THEN THE ROUTINE WILL
     64                                                                 ;HALT.  PROCEEDING FROM THE "HALT"
     65                                                                 ;WILL RERUN THE RXBOOT.
```

```
                                                        ;CONFIGURATION
                                                        ;
173102                          CONFIG:
173102  112737  000012  177524        MOVB    #FIVE,@#LIGHTS  ;DIAGNOSTIC LIGHT DISPLAY=5

173110  012704                        MOV     (PC)+,R4        ;R4=COMMAND WORD
173112  000001                        .WORD   00!1            ;UNIT 0 !  FUNCTION ! GO

173114  005700                        TST     R0              ;SET CORRECT BIT FOR DRIVE SELECTED
173116  001402                        BEQ     10$             ;BRANCH IF DRIVE 0
173120  052704                        BIS     (PC)+,R4        ;SET BIT4 IF DRIVE 1
173122  000020                        .WORD   BIT4

173124                          10$:
173124  032737  000100  177524        BIT     #BIT6,@#SWITCH  ;READ SWITCH A7 ON BVD11-AA MODULE
173132  001402                        BEQ     20$             ;A7=1 RX02      DO NOT BRANCH
173134  052704                        BIS     (PC)+,R4        ;A7=0 RX01      DO BRANCH
173136  000400                        .WORD   BIT8            ;IF RX02 SET DENSITY BIT 8 IN

173140  000207                  20$:  RTS     PC              ;COMMAND WORD
                                                        ;EXIT RETURN TO CALLER OF CONFIG
```

```
                                                ;READ AND WRITE ROUTINE
                                                        ;ENTER THIS ROUTINE WITH R4 FUNCTION
                                                        ;CODE SET TO READ OR WRITE
                                                        ;TRACK 1 SECTOR 1

173142  012702  000010          RXRDWR: MOV     #10,R2          ;SET ERROR RETRY

173146  004767  000176          10$:    JSR     PC,RXGO         ;PERFORM COMMAND IN R4
173152  103003                  15$:    BCC     20$             ;RETURN WITH CARRY = 0 INDICATING NO ERROR
173154  077204                          SOB     R2,10$          ;CARRY = 1 ERROR.  DO 10 RETRYS BEFORE HALTING
                                                ;*****************************************************
173156  004767  000320                  JSR     PC,RXERR        ;ERROR AFTER 10 ATTEMPTS TO DO FUNCTION IN R4.
                                                        ;PROCEED WILL CAUSE RXBOOT TO BE RERUN
                                                        ;R0 = DRIVE SELECTED
                                                        ;R1 = CSR ADDRESS
                                                        ;R2 = ERROR RETRY COUNTER = 000 000
                                                        ;R3 = CONTENTS OF CSR
                                                        ;R4 = COMMAND WORD
                                                        ;R5 = ERROR STATUS OF CONTROLLER
                                                ;*****************************************************

173162  032711  000040          20$:    BIT     #BIT5,(R1)      ;TEST FOR DONE

173166  001006                          BNE     30$             ;BRANCH IF DONE SET
173170  112761  000001  000002          MOVB    #1,2(R1)        ;LOAD THE TRACK AND SECTOR WITH 1
173176  004767  000150                  JSR     PC,RXGO1        ;DONE NOT SET BUT TR NEED TO BE TESTED.
173202  000763                          BR      15$             ;GO CHECK FOR ERRORS

173204  042704                  30$:    BIC     (PC)+,R4        ;DONE SET AND NO ERROR
173206  000016                          .WORD    BIT1! BIT2! BIT3

173210  000207                          RTS     PC              ;CLEAR FUNCTION BITS AND RETURN
```

```
                                                        ;FILL OR EMPTY FUNCTION
                                                        ;R4 IS SETUP BEFORE CALL IS MADE
                                                        ;R4 = FILL FUNCTION = 000
                                                        ;R4 = EMPTY FUNCTION = 001
                                                        ;R5 IS SETUP BEFORE CALL IS MADE
                                                        ;R5 IS BUS ADDRESS
                                                        ;R2 IS SETUP BEFORE CALL IS MADE
                                                        ;R2 IS RETRY COUNTER
173212                          FILEPT:
173212  012702  000010                  MOV     #10,R2          ;ERROR RETRY COUNTER
173216  004767  000126          5$:     JSR     PC,RXGO         ;PERFORM FILL OR EMPTY FUNCTION
173222  103003                          BCC     10$             ;RETURN WITH CARRY = 0 INDICATING NO ERROR
173224  077204                          SOB     R2,5$           ;
                                                        ;ERROR AFTER 10 TRYS TO DO FILL OR EMPTY
173226  004767  000250                  JSR     PC,RXERR        ;PROCEED WILL CAUSE RXBOOT TO BE RERUN
                                                        ;R0 = DRIVE SELRXTED
                                                        ;R1 = CSR ADDRESS
                                                        ;R2 = ERROR RETRY COUNTER
                                                        ;R3 = CONTENTS OF CSR.
                                                        ;R4 = COMAND WORD.
                                                        ;R5 = ERROR STATUS REGISTER.
173232  032711  004000          10$:    BIT     #BIT11,(R1)     ;TEST FOR RX01 OR RX02

173236  001020                          BNE     40$             ;BIT 11 = 0 THEN RX01. DO NOT BRANCH.
                                                        ;BIT 11 = 1 THEN RX02. DO BRANCH.
                                                ;RX01 BEING USED
173240  032704  000002          15$:    BIT     #BIT1,R4        ;TEST IF FILL BIT1 = 0. DO NOT BRANCH.
173244  001003                          BNE     20$             ;EMPTY BIT1 = 1. DO BRANCH.

                                                        ;FILL
173246  112561  000002                  MOVB    (R5)+,2(R1)     ;FILL FUNCTION
173252  000402                          BR      30$

                                                        ;EMPTY
173254  116125  000002          20$:    MOVB    2(R1),(R5)+     ;EMPTY FUNCTION
173260  004767  000066          30$:    JSR     PC,RXGO1        ;CHECK FOR TRANSFER REQUEST
173264  103001                          BCC     35$             ;IF CARRY=0 NO ERROR BRANCH
173266  000753                          BR      5$              ;IF CARRY = 1 ERROR REDO ENTIRE COMMAND

173270  032711  000040          35$:    BIT     #BIT5,(R1)      ;CHECK FOR DONE WHICH WILL INDICATE
173274  001761                          BEQ     15$             ;FUNCTION COMPLETED, BRANCH IF DONE NOT SET.
173276  000421                          BR      60$             ;GO TO COMMON EXIT
                                                ;RX02 BEING USED

173300  012761  000200  000002  40$:    MOV     #200,2(R1)      ;LOAD WORD COUNT
173306  004767  000040                  JSR     PC,RXGO1        ;LOAD WORD COUNT AND WAIT FOR T/R.
173312  103001                          BCC     45$             ;IF CARRY=0 NO ERROR DO BRANCH
173314  000740                          BR      5$              ;IF CARRY = 1 ERROR BE ENTIRE COMMAND
173316  010561  000002          45$:    MOV     R5,2(R1)        ;LOAD BUS ADDRESS
173322  004767  000024                  JSR     PC,RXGO1        ;WAIT FOR T/R AND DONE
```

```
     58 173326  103001                    BCC     50$             ;IF CARRY = 0 NO ERROR, DO BRANCH
     59 173330  000732                    BR      5$              ;IF CARRY = 1 ERROR REDO ENTIRE FUNCTION
     60 173332  032711  000040     50$:    BIT     #BIT5,(R1)      ;'DONE' BIT SHOULD BE SET
     61 173336  001001                    BNE     60$             ;IF IT IS SET, DO END OF FUNCTION
     62 173340  000724                    BR      FILEPT          ;IF NOT SET ERROR REDO ENTIRE FUNCTION.
     63
     64 173342  042704              60$:  BIC     (PC)+,R4        ;CLEAR FUNCTION BITS
     65 173344  000016                    .WORD   BIT1!BIT2!BIT3
     66
     67 173346  000207                    RTS     PC              ;RETURN
     68                                                           ;RETURN WITH CARRY=1 INDICATES ERROR
     69                                                           ;       CARRY =0 NO ERROR
     70                                                           ;R0= DRIVE SELECTED
     71                                                           ;R1 = CSR ADDRESS
     72                                                           ;R2 = ERROR RETRY COUNTER
     73                                                           ;R3 = WATCH DOG TIMER (WITH NO ERROR FLAG)
     74                                                           ;R3 = CONTENTS OR CSR (WITH A ERROR FLAG)
     75                                                           ;R4 = COMMAND WORD
     76                                                           ;R5 = ERROR STATUS REGISTER (WITH A ERROR FLAG)
```

```
                                                        ;RXGO
                                                                ;LOOK FOR "TRANSFER REQUEST" OR DONE.
173350  010411                  RXGO:   MOV     R4,(R1)         ;EXECUTE FUNCTION-GO-
173352  005003                  RXGO1:  CLR     R3              ;SETUP WATCH DOG TIMER
173354  105711                  10$:    TSTB    (R1)            ;CHECK FOR "TRANSFER REQUEST"
173356  100410                          BMI     20$             ;BRANCH IF SET
173360  032711  000040                  BIT     #BIT5,(R1)      ;CHECK FOR "DONE"
173364  001005                          BNE     20$             ;BRANCH IF SET
173366  005203                          INC     R3              ;NEITHER "TRANSFER REQUEST" OR "DONE" SET.
173370  001371                          BNE     10$             ;DUMP TIMER AND IF IT DOES NOT OVERFLOW RETEST
                                                        ;*****************************************************
173372  011103                          MOV     (R1),R3         ;WATCHDOG TIMER ERROR
173374  004767  000102                  JSR     PC,RXERR        ;TIME HAS EXPIRED AND NO "DONE" BIT
                                                                ;SET BY CONTROLLER IN TIME ALLOTED
                                                                ;POSSIBLE PROBLEM
                                                                ;       NO POWER TO FLOPPY
                                                                ;       CONTROLLER IS DEAD
                                                                ;ODT 'PROCEED WILL RESTART FLOPPY BOOT
                                                                ;R0 = SELECTED FLOPPY DRIVE
                                                                ;R1 = ADDRESS OF CSR
                                                                ;R2 = ERROR RETRY COUNTER
                                                                ;R3 = CONTENTS OF CSR
                                                                ;R4 = COMMAND WORD
                                                        ;*****************************************************
173400  005711                  20$:    TST     (R1)            ;TEST IF ERROR BIT SET
173402  100014                          BPL     RXGOBY          ;BRANCH IF NO ERROR.  TST CLEARS CARRY

                                                        ;ERROR BIT SET
173404  010405                  RXSTAT: MOV     R4,R5           ;SET COMMAND WORD TO R5
173406  042705                          BIC     (PC)+,R5        ;CLEAR FUNCTION BITS
173410  000016                          .WORD   BIT1!BIT2!BIT3

173412  052705                          BIS     (PC)+,R5        ;SET FUNCTION TO DO A READ ERROR STATUS
173414  000012                          .WORD   BIT1!BIT3

173416  010511                          MOV     R5,(R1)         ;EXECUTE COMMAND "READ ERROR STATUS"

173420  032711  000040          25$:    BIT     #BIT5,(R1)      ;TEST FOR DONE, THIS COMMAND TAKES

173424  001775                          BEQ     25$             ;250 MSEC TO COMPLETE.
173426  016105  000002          30$:    MOV     2(R1),R5        ;LOAD R3 WITH CONTENTS OF CSR
173432  000261                          SEC                     ;LOAD R5 WITH ERROR STATUS REGISTER.

173434  011103                  RXGOBY: MOV     (R1),R3         ;SET CARRY IF ERROR
173436  000207                          RTS     PC              ;LOAD CONTENTS OF CSR INTO R3
                                                                ;EXIT                   R0=SELECTED RX DRIVE
                                                                ;CARRY = 0 NO ERROR     R1=ADDRESS OF CSR
                                                                ;CARRY = 1 ERROR        R2=ERROR RETRY COUNTER
                                                                ;                       R3=NO ERROR WATCHDOG, TIME
                                                                ;                       R3=ERROR CONTENTS OF CSR
                                                                ;                       R4=COMMAND WORD
                                                                ;                       R5=ERROR REGISTER
```

```
                                                        ;TEST FOR NOP IN SECONDARY BOOTSTRAP.
                                        TST240:
173440                                                  
173440  112737  000011  177524          MOVB    #SIX,@#LIGHTS   ;DIAGNOSTIC LIGHT DISPLAY = 6
173446  022737  000240  000000          CMP     #240,@#0        ;TEST LOCATION ZERO FOR A "NOP" (240)
173454  001402                          BEQ     10$             ;BRANCH IF NOP THERE
                                                        
                                                ;***************************************************************************
173456  004767  000020                  JSR     PC,RXERR        ;SECONDARY BOOTSTRAP REQUIRES A
                                                        ;NOP IN LOCATION 0.  NONE THERE
                                                        ;PROCEED WILL RERUN BOOTSTRAP
                                                ;***************************************************************************
                                                        
173462  112737  000037  177524  10$:    MOVB    #BLANK,@#LIGHTS ;CLEAR DIAGNOSTIC LIGHT DISPLAY
173470  005005                          CLR     R5              ;START AT LOCATION 0
173472  012704  000400                  MOV     #PAGE0,R4       ;PAGE RE-SET TO ZERO
173476  000167  000264                  JMP     CPAG13          ;BYE
```

```
                                                        ;RX ERROR TEST ROUTINE
                                                                ;TEST IF PROGRAM IS TO "HALT" OR JUMP TO "EPROM".
173502                                  RXERR:
173502  032737  003400  177524          BIT     #BIT8!BIT9!BIT10,@#SWITCH
                                                                ;TEST FOR THE "ROM BOOT" .
173510  001006                          BNE     20$             ;BRANCH IF ROM SELLECTED.
                                                                ;BIT8= ROM BOOTSTRAP ON BVD11-AA BOARD.
                                                                ;BIT9= EPROM SELECTED
                                                                ;BIT10= SYSTEM ROM

                                                        ;*******************************************************************
173512  004767  177666          10$:    JSR     PC,RXSTAT       ;READ DRIVE STATUS INTO R5.
173516  011602                          MOV     (SP),R2         ;MAKE R2 ERROR POINTER.


173520  000000                          HALT                    ;RX  DEVICE ERROR.



                                                                ;THE ERROR WAS CAUSED BY A RX LOAD DEVICE.
                                                                ;STACK POINTS TO ROUTINE THAT CALLED ERROR "HALT".
173522  000167  177266                  JMP     RXREDO          ;PROCEEDING FROM THIS "HALT" WILL RERUN THE
                                                                ;RX BOOTSTRAP. REFER TO SPECIFIC ERROR
                                                                ;FOR CAUSE OF FAILURE.
                                                                ; R0 = DRIVE SELECTED
                                                                ; R1 = ADDRESS OF CSR
                                                                ; R2 = ADDRESS OF ERROR CALL.
                                                                ; R3 = CONTENTS OF CSR
                                                                ; R4 = COMMAND WORD
                                                                ; R5 = DRIVE ERROR STATUS REGISTER

                                                        ;*******************************************************************
173526                          20$:
173526  012704  007416                  MOV     #PAGE16,R4      ;PAGE FOR ROM BOOTSTRAP
173532  012705  173000                  MOV     #ROBOOT,R5      ;ADDRESS OF ROM BOOTSTRAP
173536  000513                          BR      CPAG13          ;GO CHANGE PAGE FOR ERROR MESSAGE SUPPLIED
                                                                ;BY A USER.
```

```
                                                        ;SET MEMORY HITH DATA SELECTED BY A USER.
                                                        ;LOAD
                                                        ;LOCATIONS 200-376 WITH THE DATA
                                                        ;PATTERN SELECTED BY THE USER.
                                                        ;USER PLACED PATTERN IN R5
                                                        ;
                                                        ;LOCATIONS 000-176 WILL BE FILLED
                                                        ;WITH THE ONE'S COMPLEMENT OF THE
                                                        ;DATA IN R5.
173540                          SETMEM:
173540  005005                          CLR     R5      ;DESIRED PATTERN IN MEMORY FOR
                                                        ;FILLING AND WRITING TO DISK
173542  000000                          HALT            ;CAUTION.  THIS PROGRAM WILL DESTROY
                                                        ;TRACK 1 SECTOR 1
173544  012703  000200                  MOV     #200,R3 ;SWITCHING COUNTER
173550  012704  001000                  MOV     #1000,R4 ;UPPER ADDRESS

173554  010544                  10$:    MOV     R5,-(R4) ;LOAD DATA IN R5 INTO ADDRESS IN R4
173556  005303                          DEC     R3      ;DECREMENT SWITCHING COUNTER
173560  001375                          BNE     10$     ;BRANCH IF MORE DATA TO WRITE
173562  005105                          COM     R5      ;SWITCHING COUNTER SET. COMPLEMENT R5
173564  010544                  20$:    MOV     R5,-(R4) ;AND FILL REST OF MEMORY UNTIL LOC. 0
173566  005704                          TST     R4      ;TEST THE ADDRESS POINTER END WHEN = 0.
173570  001375                          BNE     20$     ;IS REACHED THEN STOP.
173572  000207                          RTS     PC      ;RETURN TO CALLER
```

```
                                        ;MAINTENANCE MODE
173574                          RXHELP:
173574  005000                  CLR     R0              ;MAKE DRIVE SELECTED = 0.
173576  012701                  MOV     (PC)+,R1        ;R0 = DRIVE SELECTED.
173600  177170                  .WORD   177170          ;R1 = ADDRESS OF CSR

173602  012706  001100          MOV     #1100,SP        ;STACK = 1100


                                        ;SET MEMORY DATA PATTERN

173606  004767  177726          JSR     PC,SETMEM       ;HALT WILL OCCURE.
                                                        ;SET R5 = TO DESIRED PATTERN TO BE WRITTEN.

173612  004767  177264          JSR     PC,CONFIG       ;SET R4 TO CORRECT COMMAND WORD.

                                        ;SETUP FOR FILL

173616  012705  000400          MOV     #400,R5         ;START FILLING CONTROLLER AT MEMORY LOC. 200

173622  004767  177364          JSR     PC,FILEPT       ;EXECUTE FILL COMMAND IN R4.


                                        ;EMPTY

173626  052704                  BIS     (PC)+,R4        ;FUNCTION BITS FOR EMPTY
173630  000002                  .WORD   BIT1

173632  005005                  CLR     R5              ;R5 IS POINTER TO START LOADING MEMORY

173634  004767  177352          JSR     PC,FILEPT       ;PERFORM FUNCTION

173640  000000                  HALT                    ;END OF MAINTENANCE MODE

                                ;PROCEED WILL WRITE TO DISK IN MAINTENANCE MODE.
```

```
                                                ;WRITE MAINTENANCE MODE ONLY
                                                ;WRITE TO DISK CYL 0 SECTOR 0 HEAD 0
173642  004767  177672          WRITE:  JSR     PC,SETMEM       ;STOP AND SET A NEW DATA PATTERN IN R5.
                                                ;WRITE REQUIRES THE FOLLOWING SET-UP
173646  004767  177230                  JSR     PC,CONFIG       ;SET R4 COMMAND WORD SETMEM DESTROYED IT.
173652  152704                          BISB    (PC)+,R4        ;COMMAND TO WRITE IN R4
173654  000004                          .WORD   BIT2            ;SET WRITE BITS.


173656  004767  177260                  JSR     PC,RXRDWR       ;EXECUTE FUNCTION
173662  103002                          BCC     30$             ;CARRY = 0 NO ERROR
                                                ;CARRY = 1 ERROR
                                        ;***************************************************************
173664  000000                          HALT                    ;WRITE ERROR
173666  000765                          BR      WRITE           ;PROCEED WILL RE RUN WRITE ROUTINE
                                                ;R0 = DRIVE UNIT
                                                ;R1 = CSR ADDRESS
                                                ;R2 = ERROR RETRY COUNTER
                                                ;R3 = CONTENTS OF CSR
                                                ;R4 = COMMAND WORD
                                                ;R5 = DRIVE STATUS
                                        ;***************************************************************
173670  000000                  30$:    HALT                    ;WRITE COMPLETED JUMP TO RXBOOT TO READ DATA JUST WRITTEN.
173672  000167  177102                  JMP     RXBOOT          ;READ THE DATA .
```

CHANGE PAGE ROUTINE FOR PAGE THIRTEEN.

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE THIRTEEN.
                                ;* CHANGE PAGE ROUTINE FOR THIS PAGE.
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;*
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*

                                                        ;FILL REMAINDING ROM LOCATIONS WITH HALTS

 173676  000000                .WORD   000000
 173700  000000                .WORD   000000
 173702  000000                .WORD   000000
 173704  000000                .WORD   000000
 173706  000000                .WORD   000000
 173710  000000                .WORD   000000
 173712  000000                .WORD   000000
 173714  000000                .WORD   000000
 173716  000000                .WORD   000000
 173720  000000                .WORD   000000
 173722  000000                .WORD   000000
 173724  000000                .WORD   000000
 173726  000000                .WORD   000000
 173730  000000                .WORD   000000
 173732  000000                .WORD   000000
 173734  000000                .WORD   00000C
 173736  000000                .WORD   000000
 173740  000000                .WORD   000000
 173742  000000                .WORD   000000
 173744  000000                .WORD   000000
 173746  000000                .WORD   000000
 173750  000000                .WORD   000000
 173752  000000                .WORD   000000
 173754  000000                .WORD   000000
 173756  000000                .WORD   000000
 173760  000000                .WORD   000000
 173762  000000                .WORD   000000
 173764  000000                .WORD   000000
         173766                .=173766
 173766  010437  177520        CPAG13: MOV     R4,@#PAGELO             ; LOAD NEW PAGE WORD.
 173772  010507                        MOV     R5,PC                   ; JAM NEW PC.

 173774  000000                        .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE

 173776  000000                        .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
                                                                        ; BY ROM BLASTING PROGRAM.

         000001'                .END
```

TABLE OF CONTENTS

  1- 10 SERIAL LINE UNIT BOOTSTRAPS
  2- 1 BASIC DEFINITIONS
  3- 1 LOCAL DATA
  4- 1 SPECIAL EQUALITIES FOR BVD11-AA
  5- 1 SLU BOOTSTRAP
  7- 1 SLU SETUP
  8- 1 SET BOOT DEVICE ADDRESS AND DEVICE CODE
  9- 1 REQUEST SECONDARY PROGRAM
 10- 1 SEND A BLOCK ON THE LINK
 11- 1 RECEIVE A MESSAGE ROM THE LINK.
 12- 1 RECEIVE A BLOCK FROM THE LINK
 13- 1 DECNET BOOT REQUEST MESSAGES
 14- 1 FLOATING DEVICE MODULO TABLE
 15- 1 CHANGE PAGE ROUTINE FOR PAGE FIFTEEN.


```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS

173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550

000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          . = 173000
```

```
                        .SBTTL  SLU BOOTSTRAP
                        ;*************************************************************************************************
                        ;* DOWNLINE LOADING BOOTSTRAP FOR THE BVD11-AA SYSTEMS WITH DECNET SOFTWARE.
                        ;*
                        ;* THIS PROGRAM LOADS THE COMPUTER MEMORY OVER A COMMUNICATION LINE USING THE
                        ;* MOP (MAINTENANCE OPERATION PROTOCOL). CURRENT MOP VERSION 1.2 JULY 1976
                        ;* MOP OPERATES WITHIN THE THE DDCMP MAINTENANCE ENVELOPES.
                        ;* CURRENT DDCMP VERSION 3.03 DECEMBER 1975
                        ;*
                        ;* DDCMP MAINTENANCE ENVELOPE MESSAGE ARE OF THE FORM
                        ;* DLE, CNT, Q, S, FILL, FILL, ADDR, CRC1, DATA, CRC2
                        ;*
                        ;* DLE - THE DDCMP MAINTENANCE HEADER CHARACTER - OCTAL 220 (8 BITS)
                        ;* CNT - THE MAINTENANCE MESSAGE COUNT FIELD- DATA LENGTH (14 BITTS)
                        ;* Q   = THE QSYNC BIT  LINK CONTROL ALWAYS  = 1.(1 BIT)
                        ;* S   = THE SELECT BIT LINK CONTROL ALWAYS = 1  (1 BIT)
                        ;* FILL         FILL CHARACTER = 000  (8 BITS)
                        ;* FILL
                        ;* ADDR         STATION ADDRESS = 1 FOR POINT TO POINT.
                        ;* CRC1         CRC-16 ON DLE TO ADDR.
                        ;*
                        ;* CODE = 10    MOP CODE TO REQUEST SECONDARY BOOT.
                        ;* DEV = 2, 4           DEVICE TYPE 2 = DU
                        ;*                      DEVICE TYPE 4 = DL
                        ;* STADR                STATION ADDRESS POINT TO POINT = 1.
                        ;* CODE = 0             PROGRAM TYPE SECONDARY LOAD.
                        ;*
                        ;* MAINTENANCE MESSAGES ARE PRECEED BY SYNC SEQUENCES APPROPRIATE TO THE
                        ;* CHARACTERTICS OF THE LINK USED. SYNCHRONOUS= 266  ASYNCHRONOUS= 377
                        ;*
                        ;*
                        ;* DEVICES SUPPORTED
                        ;* DUV11, DUPV11, DLV11E, OR C, D, WIRED AS E.
                        ;*
                        ;* DECNET REGISTER USAGE
                        ;* R0 -- DEVICE INDEX CODE
                        ;* R1 - DEVICE CSR ADDRESS
                        ;*
                        ;* DEVICE TYPE CODES
                        ;* DU11 - 2
                        ;* DL11 - 4
                        ;* DUP11 - 10
                        ;*
```

```
                        ;* DIAGNOSTIC LIGHT DISPLAY FOR SLU BOOTSTRAP
                        ;*      5 HARDWARE ERROR
                        ;*              INCORRECT SWITCH SETTING ON M 8012 TERMINATOR BOARD. (PROGRAM WILL HALT)
                        ;*
                        ;*              ERROR IN CSR ADDRESS (FLOATING SPACE AREA).
                        ;*
                        ;*              NO DSR SET. (DATA SET READY)
                        ;*
                        ;*              NO CTS SET. (CLEAR TO SEND)
                        ;*
                        ;*              PROGRAM WILL LOOP WITH THIS LIGHT ON UNTIL DSR OR CTS IS SET.
                        ;*
                        ;*              INDICATES A RETRANSMITTION OF REQUEST MESSAGE CAUSED BY A ERROR.
                        ;*
                        ;*      6 RECEIVED HEADER OK,  NOW RECEIVING DATA.
                        ;*
                        ;*              THE DDCMP HEADER WAS CORRECT AND THE PROGRAM IS NOW CHECKING
                        ;*              FOR THE DATA PORTION OF THE MESSAGE BLOCK.THIS LIGHT WILL
                        ;*              STAY ON UNTIL THE RECEIVE DONE FLAG IS SET.
                        ;*
                        ;*      7  NO RESPONCE FROM HOST
                        ;*              A MESSAGE WAS SENT TO THE HOST BUT NO RESPONCE WAS
                        ;*              RECEIVED. THE PROGRAM WILL RETRANSMITT THE MESSAGE
                        ;*              AFTER A PERIOD OF TIME.
                        ;*
                        ;*      10 DATA RECEIVED DONE BIT SET.
                        ;*              THE RECEIVE BUFFER HAS SET A FLAG INDICATING DATA HAS BEEN RECEIVED.
                        ;*
                        ;*              THIS DATA MUST BE CHECKED FOR THE CORRECT DDCMP FORMAT.
                        ;*              THIS LIGHT INDICATES DATA IS BEING RECEIVED BUT IT IS
                        ;*              NOT CORRECT. THIS LIGHT WILL CHANGE WHEN A DLE, SOH, ENQ, IS RECEIVED.
                        ;*
                        ;*      11 DATA RECEIVED WAS A DDCMP MESSAGE
                        ;*              A MESSAGE HAS BEEN RECEIVED IN DDCMP FORMAT. THIS LIGHT
                        ;*              WILL STAY ON WHILE DATA IS BEING RECEIVED. IT WILL GO OFF
                        ;*              IF THE HEADER DID NOT CONTAIN A DLE.
                        ;*
                        ;*              IF THERE WAS A CRC ERROR ON HEADER.
                        ;*
                        ;*              IF THERE WAS A CRC ERROR ON DATA.
                        ;*
                        ;*              THE PROGRAM WILL RETRAMSMITT THE MESSAGE.
                        ;*
                        ;*      SWITCH SETTINGS FOR SLU BOOTSTRAP
                        ;*              SWITCH A4       DUV-11  TEST BIT3
                        ;*              SWITCH A5       DLV-11  TEST BIT4
                        ;*              SWITCH A6       DUPV-11 TEST BIT5
                        ;*              SWITCH A7       OFF     FOR FUTURE DEVICES.
                        ;*
                        ;*
                        ;*****************************************************************************************
```

```
                        .SBTTL  SLU SETUP
                                        ; ROM PAGE IS PAGE14. 006414 IN PAGE CONTROL REGISTER.
                                                ; DIAGNOSTIC LIGHT DISPLAY = 5.
173000                  SLBOOT:
173000  112737  000012  177524          MOVB    #FIVE,@#LIGHTS  ; DIAGNOSTIC LIGHT DISPLAY = 5.

173006  012701  000010                  MOV     #BIT3,R1        ; TEST THE DIAGNOSTIC SWITCHES FOR DEVICE SELECTED
173012  012702  173042                  MOV     #SLTABL,R2      ; AND LOAD DEVICE CODE INTO R0 FOR SELECTED DEVICE
173016  030137  177524          10$:    BIT     R1,@#SWITCH     ; TEST FOR DEVICE IN SWITCHES OM BVD11-AA
173022  001014                  BNE     GOTIT           ; BRANCH WHEN DEVICE IS FOUND IN SWITCHES

173024  005722                  TST     (R2)+           ; NOT THIS SWITCH BUMP ADDRESS POINTER
173026  006301                  ASL     R1              ; AND MOVE TEST BIT IN R1.

173030  032701  000100                  BIT     #BIT6,R1        ; TEST FOR NO DEVICE SELECTED
173034  001770                  BEQ     10$             ; BRANCH IF FUTHER SWITCHES TO TEST

                                        ; ************************************************************
173036  000000                  HALT                    ; NO DEVICE SELECTED IN SWITCHES ON BVD11-AA
173040  000757                  BR      SLBOOT          ; PROCEEDING FROM HALT WILL RERUN SLBOOT
                                        ; ************************************************************
173042                  SLTABL:
173042     003     020                  .BYTE   003,16.         ; DEVICE CODES FOR DUV

173044     000     000                  .BYTE   000,000         ; DEVICE CODES FOR DLV
                                                ; ALSO ERROR HALT FOR INVALID CSR REGISTER.

173046     004     356                  .BYTE   004,-18.        ; DEVICE CODE FOR DUPV

                                                ; THIS ROUTINE REQUIRES A HALT INSTRUCTION  LOCATED FOUR
                                                ; BYTES PRIOR TO NODEV
173050                  NODEV:
173050  122243                  CMPB    (R2)+,-(R3)     ; FLOAT DEVICE INTERRUPT ROUTINE
173052  000002                  RTI                     ; RETURN
                                                ; R2 = POINTER TO DEVICE TABLE
                                                ; R3 = LOW BYTE FROM SLTABL.

173054                  GOTIT:
173054  005000                  CLR     R0              ; MAKE DEVICE UNIT = 0.
173056  011201                  MOV     (R2),R1         ; SET R1 TO CORRECT DEVICE CODE FROM TABLE.
173060  000400                  BR      NETBOT          ; GO TO THE NETWORK BOOT ROUTINE FOR SELECTING
                                                ; CORRECT DEVICE IN FLOATING SPACE.
```

```
                        .SBTTL  SET BOOT DEVICE ADDRESS AND DEVICE CODE
                                ; FIND THE DEVICE IN FLOATING SPACE IF DUV, DUPV, DLV11-E
                                ;
                                ; R0 = UNIT NUMBER (INPUT)
                                ; R0 = DEVICE CODE (OUTPUT)
                                ; R1 = DEVICE TYPE (INPUT)
                                ; R1 = DEVICE CSR  (OUTPUT)
                                ;

173062                  NETBOT:
173062  012706  017776          MOV     #17776,SP       ; SET STACK TO 4K -2.
173066  012705  000006          MOV     #6,R5           ; TRAP PS ADDRESS
173072  012715  000340          MOV     #340,(R5)       ; ALLOW NO INTERRUPTS FOR  NEW PSW.
173076  012745  173050          MOV     #NODEV,-(R5)    ; TRAP RETURN ADDRESS

173102  012702  173706          MOV     #DEVTAB,R2      ; R2 = POINTER TO DEVICE TABLE.
173106  010146                  MOV     R1,-(SP)        ; SAVE UNIT CODE
173110  012701  175610          MOV     #175610,R1      ; SET CSR FOR FOR DLV11 UNIT 0.
173114  111603                  MOVB    (SP),R3         ; SET NUMBER IN FLOATING SPACE
173116  001411                  BEQ     UNIT            ; BRANCH IF DLV11 SELECTED.

173120  012701  160010          MOV     #160010,R1      ; STARTING ADDRESS OF FLOATING ADDRESS
173124  005711          FLOAT:  TST     (R1)            ; TEST FOR DEVICE PRESENT IF NOT TRAP TO NODEV.
173126  111204                  MOVB    (R2),R4         ; MODULO INCREMENT.
173130  060401                  ADD     R4,R1           ; UPDATE ADDRESS
173132  005201                  INC     R1              ; BY MODULO
173134  040401                  BIC     R4,R1           ; IN TABLE.

173136  005703                  TST     R3              ; IS THIS DEVICE NUMBER?
173140  001371                  BNE     FLOAT           ; IF R3 NOT ZERO NOT DEVICE NUMBER.

                        ; ADD UNIT DISPLACEMENT TO UNIT 0 CSR ADDRESS

173142                  UNIT:
173142  160515                  SUB     R5,(R5)         ; SET TRAP AT 4 TO A HALT
173144  006300                  ASL     R0              ; UNIT NUMBER TIMES 2
173146  006300                  ASL     R0              ; UNIT NUMBER TIMES 4
173150  006300                  ASL     R0              ; UNIT NUMBER TIMES 8
173152  060001                  ADD     R0,R1           ; CSR ADDRESS + UNIT*8.
173154  000316                  SWAB    (SP)            ; DEVICE CODE TO LOW BYTE
173156  112600                  MOVB    (SP)+,R0        ; DEVICE TYPE IN R0.
                                                        ; CSR ADDRESS IN R1.
```

```
                                .SBTTL  REQUEST SECONDARY PROGRAM

                                ;* REQUEST SECONDARY PROGRAM FROM HOST
                                ;* MESSAGE SENT IS
                                ;*
                                ;* CODE 8 = REQUEST SECONDARY PROGRAM PRIMARY MODE RUNNING
                                ;* CODE (1), DEVICE TYPE(1), STATION ADDRESS(1), DEATURES(1)
                                ;*
                                ;* R0 = DEVICE INDEX
                                ;* R1 = RECEIVE CSR ADDRESS (AFTER SEND1 CALL R1= TRANSMITT CSR ADDRESS)
                                ;* R3 = BUFFER LENGTH
                                ;* R4 = BUFFER ADDRESS.

173160  012706  017460        SNDREQ: MOV     #17400+<8.*6.>,SP
                                                        ; SET STACK ADDRESS = 17400 + 8 TIMES LOOP DEC.
173164  012704  173644        SNDRQ1: MOV     #DLREQ,R4       ; DLV-11 REQUEST MESSAGE.

173170  012737  000012  177524        MOV     #FIVE,@#LIGHTS  ; INDICATE A RETRANSMITTION REQUESTED
173176  060004                        ADD     R0,R4           ; SET R4 TO CORRECT DEVICE MESSAGE ADDRESS.
173200  112403                        MOVB    (R4)+,R3        ; MESSAGE LENGTH AND PADS IN R3.
```

SEND A BLOCK ON THE LINK

```
                                .SBTTL  SEND A BLOCK ON THE LINK
                                ;* R0 = DEVICE TYPE CODE
                                ;* R1 = RECEIVE CSR ADDRESS.(AFTER CALL R1 POINTS TO TRANSMITT CSR).
                                ;* R3 = BUFFER LENGTH.
                                ;* R4 = BUFFER ADDRESS.

173202  012711  000006                MOV     #6,(R1)         ;SET DTR AND RTS
173206  012761  036226  000002        MOV     #36000+SSYN,2(R1)
                                                              ; SET DUV11 SYNC REGISTER THIS IS A NOP ON DLV11

173214  105700                        TSTB    R0              ; IS IT A DUPV-11 OR DLV-11
173216  001407                        BEQ     1$              ; BRANCH IF A DLV-11
173220  100003                        BPL     2$              ; NO DUV-11 PARCSR OK.
173222  012761  101226  000002        MOV     #101000+SSYN,2(R1)
                                                              ; SET FOR DUPV-11

173230  032711  001000          2$:   BIT     #1000,(R1)      ; TEST FOR DSR
173234  001775                        BEQ     2$              ; NOT SET YET

173236  032711  020000          1$:   BIT     #20000,(R1)     ; TEST FOR CTS
173242  001775                        BEQ     1$              ; NOT SET YET

173244  022121                        CMP     (R1)+,(R1)+     ; SET R1 TO TRANSMITT CSR

173246  052721  000030                BIS     #30,(R1)+       ; HDX AND SEND ON NOP IF DLV-11.
173252  012711  000626                MOV     #400+SSYN,(R1)  ; START IT WITH TSOM.
173256  000401                        BR      STEST           ; TEST FOR DONE

173260  112411                SEND:   MOVB    (R4)+,(R1)      ; MOVE DATA BYTE TO DEVICE TRANSMIT BUFFER
173262  105761  177776        STEST:  TSTB    -2(R1)          ; TEST FOR DONE
173266  100375                        BPL     STEST           ; NOT YET
173270  005303                        DEC     R3              ; DECREMENT COUNTER
173272  001372                        BNE     SEND            ; MORE TO SEND
```

```
                                .SBTTL  RECEIVE A MESSAGE ROM THE LINK.

                                ;* CODE SHOULD BE
                                ;*      0 = PROGRAM LOAD WITH TRANSFER ADDRESS
                                ;*
                                ;*      R0 = DEVICE INDEX
                                ;*      R1 = RECEIVE CSR ADDRESS (AFTER RESET FROM TRANSMITT DONE)
                                ;*      R2 = BUFFER START       (OUTPUT)
                                ;*      R3 = BUFFER LENGTH
                                ;*      R4 = BUFFER ADDRESS (NEXT LOCATION).
                                ;*      R5 = PARTIAL CRC
                                ;*

173274                          GETMSG:
173274  112737  000010  177524          MOVB    #SEVEN,@#LIGHTS
                                                        ; TURN ON LIGHT 7 INDICATING
                                                        ; MESSAGE SENT AND WAITING FOR RESPONCE FROM HOST.
173302  042741  000020                  BIC     #20,-(R1)               ; DROP SEND ON DUV11  A NOP ON DLV.
173306  024141                          CMP     -(R1),-(R1)             ; RESET R1 TO RECEIVE CSR
173310  005004                          CLR     R4                      ; BUFFER ADDRESS
173312  012703  000010                  MOV     #8.,R3                  ; R3 = HEADER LENGTH.
173316  004767  000062                  JSR     PC,RECV1                ; GET THE HEADER

173322  001316                          BNE     SNDREQ                  ; BRANCH IF BAD CRC ON HEADER

173324  122527  000220                  CMPB    (R5)+,#DLE              ; IS IT A DLE  MESSAGE IN LOCATION 0?
173330  001313                          BNE     SNDREQ                  ; NO

173332  113703  000002                  MOVB    @#2,R3                  ; HIGH BYTE COUNT
173336  042703  177700                  BIC     #177700,R3              ; CLEAR FLAGS AND OTHER BYTE
173342  000303                          SWAB    R3                      ; SWAP BYTES
173344  151503                          BISB    (R5),R3                 ; LOW BYTE COUNT LOCATION 1
173346  122323                          CMPB    (R3)+,(R3)+             ; ADD TWO FOR CRC
173350  005004                          CLR     R4                      ; BUFFER ADDRESS
173352  112737  000011  177524          MOVB    #SIX,@#LIGHTS           ; INDICATE DATA PORTION OF MESSAGE BEING RECEIVED

173360  004767  000032                  JSR     PC,RECV                 ; GET DATA FIELD
173364  001275                          BNE     SNDREQ                  ; CRC ERROR ON DATA
173366  005715                          TST     (R5)                    ; CHECK FOR CODE 0 IN LOCATION 0.
173370  001273                          BNE     SNDREQ                  ; NOT CORRECT
173372  112737  000037  177524          MOVB    #BLANK,@#LIGHTS
                                                        ; TURN OFF LIGHTS BEFORE JUMPING TO CODE.
173400  000137  000006                  JMP     @#6                     ; START CODE RECEIVED.
```

```
                        .SBTTL  RECEIVE A BLOCK FROM THE LINK
                        ;* RECEIVE A BLOCK FROM THE LINK.
                        ;* OPTIONALLY INITIALIZE THE LINK.
                        ;* EACH ENTRY CLEARS AND COMPUTES CRC.
                        ;* ONLY FULL HEADERS OR DATA FIELDS MAY BE RECEIVED AT A TIME.
                        ;*
                        ;* R0 = DEVICE TYPE CODE
                        ;* R1 = RECEIVE CSR ADDRESS
                        ;* R2 = SCRATCH
                        ;* R3 = BUFFER LENGTH
                        ;* R4 = BUFFER ADDRESS (NEXT LOCATION)
                        ;* R5 = PARTIAL CRC
173404                  RECV1:
173404  042711  000024          BIC     #24,(R1)        ; CLEAR RTS AND SEARCH SYNC . NOP ON DLV11.
173410  012711  000422          MOV     #422,(R1)       ; SET SEARCH,STRIP,DTR  NOP ON DLV11
173414  012746                  MOV     (PC)+,-(SP)     ; DLV-11 SEARCH/STRIP FLAG
173416                  RECV:
173416  005046                  CLR     -(SP)           ; DLV11 NO STRIP
173420  005005                  CLR     R5              ; INITIALIZE CRC
173422                  RTEST:
173422  012702  000017          MOV     #15.,R2         ; LONG LOOP VALUE
173426  005046                  CLR     -(SP)           ; SHORT LOOP
173430                  2$:
173430  105711                  TSTB    (R1)            ; TEST FOR DEVICE DONE
173432  100416                  BMI     RDONE           ; ALL DONE
173434  005316                  DEC     (SP)            ; DECREMENT SHORY COUNTER
173436  001374                  BNE     2$              ; TEST AGAIN

173440  005302                  DEC     R2              ; DECREMENT LONG COUNTER
173442  001372                  BNE     2$              ; AND TEST AGAIN
173444  105706                  TSTB    SP              ; CHECK STACK AT OR BELOW 17400
173446  003246                  BGT     SNDRQ1          ; LOOP ONCE MORE (8 TIMES TOTAL)
173450  005011                  CLR     (R1)            ; DROP DTR - HANG UP
173452                  HNGLOP:
173452  012703  000012          MOV     #10.,R3         ; LONG LOOP COUNTER
173456  005302          4$:     DEC     R2              ; DECREMENT SHORT COUNTER
173460  001376                  BNE     4$              ; DO KILL TIME
173462  005303                  DEC     R3              ; DECREMENT LONG COUNTER
173464  001374                  BNE     4$              ; AGAIN
173466  000634                  BR      SNDREQ          ; HNG UP LONG ENOUGH- ANSWER AGAIN
173470                  RDONE:
173470  112737  000007  177524  MOVB    #TEN,@#LIGHTS
                                                        ; LIGHTS INDICATING MESSAGE RECEIVED BUT
                                                        ; CHECKING FOR CORRECT MESSAGE DATA.
173476  005726                  TST     (SP)+           ; CLEAN UP STACK - LOOP CTR
173500  042711  000400          BIC     #400,(R1)       ; NO STRIP SYNC - NOP ON DLV-11
173504  116114  000002          MOVB    2(R1),(R4)      ; STORE IT
173510  105700                  TSTB    R0              ; IS IT A DLV-11
173512  001013                  BNE     1$              ; NO
```

RECEIVE A BLOCK FROM THE LINK

```
 58 173514  005716                        TST     (SP)            ; TEST SYNC MODE
 59 173516  001411                        BEQ     1$              ; NO STRIP
 60 173520  121427  000220                CMPB    (R4),#DLE       ; IS IT A MAINTENANCE MESSAGE?
 61 173524  001406                        BEQ     1$              ; YES
 62
 63 173526  121427  000005                CMPB    (R4),#ENQ       ; IS IT AN ENQ MESSAGE
 64 173532  001403                        BEQ     1$              ; YES
 65
 66 173534  121427  000201                CMPB    (R4),#SOH       ; IS IT A SOH MESSAGE
 67 173540  001330                        BNE     RTEST           ; NO NOT DDCMP MESSAGE START
 68
 69 173542                          1$:
 70 173542  112737  000006  177524        MOVB    #ELEVEN,@#LIGHTS
 71                                                              ; MESSAGE RECEIVED AND IS A DDCMP MESSAGE
 72 173550  005016                        CLR     (SP)            ; SET NO STRIP
 73 173552  112446                        MOVB    (R4)+,-(SP)     ; BYTE TO ADD
 74 173554  012702  000010                MOV     #8.,R2          ; NUMBER BITS PER BYTE
 75
 76 173560                          CRCLOP:
 77 173560  000241                        CLC                     ; CLEAR CARRY
 78 173562  006005                        ROR     R5              ; LOW BIT PARTIAL TO CARRY
 79 173564  006016                        ROR     (SP)            ; CARRY TO BYTE AND BYTE TO CARRY.
 80 173566  102006                        BVC     1$              ; XOR OF PARTIAL AND BYTE LOW BITS
 81
 82 173570  012746  120001                MOV     #POLY,-(SP)     ; XOR POLY TO PARTIAL (4 INSTRUCTIONS)
 83 173574  040516                        BIC     R5,(SP)         ; NOT PARTIAL AND POLY
 84 173576  042705  120001                BIC     #POLY,R5        ; NOT POLY AND PARTIAL.
 85 173602  052605                        BIS     (SP)+,R5        ; POLY XOR PARTIAL
 86
 87 173604                          1$:
 88 173604  005302                        DEC     R2              ; DECREMENT BIT COUNT
 89 173606  003364                        BGT     CRCLOP          ; ONCE MORE
 90 173610  005726                        TST     (SP)+           ; CLEAN UP STACK BYTE TO ADD
 91 173612  005303                        DEC     R3              ; DECREMENT BYTE COUNTER
 92 173614  003302                        BGT     RTEST           ; ONCE MORE
 93 173616  010526                        MOV     R5,(SP)+        , SET CC AND CLEAN STACK - DLV11 FLAG
 94 173620  000207                        RTS     PC              ; RETURN
```

```
                                        .SBTTL  DECNET BOOT REQUEST MESSAGES
                                        ;*
                                        ;* THESE THREE REQUEST MESSAGE MUST BE IN THIS ORDER
                                        ;*

173622                                  DUPREQ:
173622  024     226     226     .BYTE   20.,SSYN,SSYN,SSYN,DLE,4,300,0,0,1,021,120
173625  226     220     004
173630  300     000     000
173633  001     021     120
173636  010     012     001     .BYTE   10,10.,1,0,43,362       ; DUPV REQUEST MESSAGE
173641  000     043     362

173644                                  DLREQ:
173644  022     377     220     .BYTE   18.,ASYN,DLE,4,300,0,0,1,021,120
173647  004     300     000
173652  000     001     021
173655  120
173656  010     004     001     .BYTE   10,4,1,0,102,61         ; DLV REQUEST MESSAGE.
173661  000     102     061

173664                                  DUREQ:
173664  024     226     226     .BYTE   20.,SSYN,SSYN,SSYN,DLE,4,300,0,0,1,021,120
173667  226     220     004
173672  300     000     000
173675  001     021     120
173700  010     002     001     .BYTE   10,2,1,0,242,60         ; DUV REQUEST MESSAGE
173703  000     242     060
```

FLOATING DEVICE MODULO TABLE

```
                        .SBTTL  FLOATING DEVICE MODULO TABLE
                        ;
                        DEVTAB:
1 173706                
2 173706     007                .BYTE   7               ; DJ11 DEVICE MODULUS
3 173707     017                .BYTE   17              ; DH11
4 173710     007                .BYTE   7               ; DQ11
5 173711     007                .BYTE   7               ; DUV11
6 173712     007                .BYTE   7               ; DUFV11
7 173713     000                .BYTE   0               ; FILLER
```

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE FIFTEEN.
                                ;;**************************************************************************
                                ;;**************************************************************************
                                ;* CHANGE PAGE LOCATIONS FOR THIS PAGE.
                                ;*
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;*
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;*
                                ;*
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*
                                ;;**************************************************************************
                                ;;**************************************************************************
                                ; FILL THE UNUSED ROM LOCATIONS WITH HALTS
 173714  000000                 .WORD   000000
 173716  000000                 .WORD   000000
 173720  000000                 .WORD   000000
 173722  000000                 .WORD   000000
 173724  000000                 .WORD   000000
 173726  000000                 .WORD   000000
 173730  000000                 .WORD   000000
 173732  000000                 .WORD   000000
 173734  000000                 .WORD   000000
 173736  000000                 .WORD   000000
 173740  000000                 .WORD   000000
 173742  000000                 .WORD   000000
 173744  000000                 .WORD   000000
 173746  000000                 .WORD   000000
 173750  000000                 .WORD   000000
 173752  000000                 .WORD   000000
 173754  000000                 .WORD   000000
 173756  000000                 .WORD   000000
 173760  000000                 .WORD   000000
 173762  000000                 .WORD   000000
 173764  000000                 .WORD   000000
         173766                 .=173766
 173766  010437  177520         CPAG15: MOV     R4,@#PAGELO             ; LOAD THE NEW PAGE WORD
 173772  010507                         MOV     R5,PC                   ; JAM THE NEW PC

 173774  000000                         .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
 173776  000000                         .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
                                                                        ; BY ROM BLASTING PROGRAM.
         000001'                .END
```

DIGITAL EQUIPMENT CORPORATION MAYNARD, MASSACHUSETTS	TITLE	BDV11-AA DIAGNOSTIC/BOOT ROM LISTING	SHEET 108 OF 118	SIZE: K	CODE: CS	NUMBER: BDV11-A-2 / REV:

TABLE OF CONTENTS

1- 9 ROM BOOTSTRAP
2- 1 BASIC DEFINITIONS
3- 1 LOCAL DATA
4- 1 SPECIAL EQUALITIES FOR BVD11-AA
5- 1 ROM BOOTSTRAP
9- 1 CHANGE PAGE ROUTINE FOR PAGE SEVENTEEN.

```
                                ; DEFINITIONS NEEDED FOR LINKING ROM TESTS
                                
173000                          START   =       173000
173514                          SWTEST  =       173514
173550                          TABLE   =       173550
                                
000000                          .ASECT                          ; SET ABSOLUTE ADDRESSING MODE.
173000                          .= 173000
```

```
                        .SBTTL  ROM BOOTSTRAP
                        ;*****************************************************************************************
                        ;*****************************************************************************************
                        ;* ROM BOOTSTRAP
                        ;* THE ROM BOOTSTRAP WILL READ THE DATA FROM THE ROM SELECTED IN THE SWITCHES
                        ;* LOCATED ON THE BVD11-AA TERMINAL BOARD.
                        ;* THE ROM BOOTSTRAP WILL USE THE LAST 10 [OCTAL] WORDS IN MEMORY.
                        ;* THE BOOTSTRAP WILL LOOK FOR THE PROPER FORMAT WHEN LOADING
                        ;* THE ROM DATA INTO MEMORY.  THE INPUT FORMAT IS:
                        ;* BYTE 1       001                     INDICATE START OF FORMAT FILE.
                        ;* BYTE 2       000
                        ;* BYTE 3       BYTE COUNT      LOW ORDER ******(CAN NOT BE A NEGATIVE NUMBER)******
                        ;* BYTE 4       BYTE COUNT      HIGH ORDER
                        ;* BYTE 5       LOAD ADDRESS    LOW ORDER
                        ;* BYTE 6       LOAD ADDRESS    HIGH ORDER
                        ;*               DATA STARTS NOW.
                        ;*               LAST FRAME CONTAINS THE CHECKSUM
                        ;*
                        ;* ADDRESS BLOCK
                        ;* TO EXIT THE BOOTSTRAP AND BEGIN EXECUTION THE BYTE COUNT WILL
                        ;* EQUAL 6 AND THE LOAD ADDRESS WILL BE USED AS  THE JUMP TO ADDRESS.  IF
                        ;* THIS ADDRESS IS ODD THE PROGRAM WILL HALT.  CONTINUING WILL
                        ;* CAUSE THE ROM BOOTSTRAP TO RESTART.
                        ;* A CHECKSUM ERROR WILL RESULT IN A HALT, CONTINUE WILL CAUSE THE
                        ;* ROM BOOTSTRAP TO ACT AS IF NO CHECKSUM ERR OCCURRED.
                        ;* IF THERE IS NO CHECKSUM ERROR AND THE ADDRESS IS EVEN THE PROGRAM WILL START
                        ;* AT THE SPECIFIED ADDRESS IN THE LAST ROM PAGE OR MAIN MEMORY.
                        ;*               R0      =       STORAGE OF CHECKSUM
                        ;*               R1      =       ADDRESS DATA WILL BE LOADED INTO.
                        ;*               R2      =       BYTE COUNT. ******(CAN NOT BE A NEGATIVE NUMBER)******
                        ;*               R3      =       LAST BYTE READ FROM ROM.
                        ;*               R4      =       ADDRESS OF LAST DATA BYTE.
                        ;*               R5      =       INITAL LOAD ADDRESS FOR THIS BLOCK OF DATA
                        ;*                                OR JUMP ADDRESS.
                        ;*               R6      =       STACK POINTER
                        ;*
                        ;*      THE STACK POINTER IS THE OFFSET FOR THE FOLLOWING LOCATIONS.
                        ;*               (SP)    =       PAGE COUNT (200 OR 172)         (XXX 772)
                        ;*               (SP+ 2) =       ROM STRUCTURE OFFSET 10 OR 20.  (XXX 774)
                        ;*               (SP+ 4) =       REDO ROM PAGE VALUE .           (XXX 776)
                        ;*
                        ;* THE SWITCH SETTINGS FOR THE ROM BOOTSTRAP ARE:
                        ;*      SWITCH B1       =       DIAGNOSTIC ROM 2        (OCTAL 40)
                        ;*      SWITCH B2       =       2K EPROM                (OCTAL 100)
                        ;*      SWITCH B3       =       16K SYSTEM ROM          (OCTAL 200)
                        ;*      SWITCH B4       =       0 EQUALS 1K ROM
                        ;*                              1 EQUALS 2K ROM         (OCTAL 400)
                        ;*****************************************************************************************
                        ;*****************************************************************************************
```

```
                                                        ; ROM PAGE IS PAGE16. 007416 IN PAGE CONTROL REGISTER.
173000                          ROBOOT:                 ; DIAGNOSTIC LIGHT DISPLAY = 12
173000  000167  000420                JMP     SETROM          ; TURN LIGHTS ON AND FIND TOP OF MEMORY.
                                                        ; LOCATED TOP OF MEMORY
173004  005746                  FOUND:  TST     -(SP)           ; NON-EXISTANT MEMORY RETURN HERE
                                                                ; MAKE SP EQUAL TO 6 LESS THAN MAX MEMORY LIMIT.
                                                                ; SP = XXX 702
                                                        ; LOCATE CORRECT ROM PAGE TO START READING DATA FROM.
173006  012701  173454                MOV     #TABL11,R1      ; LOCATE CORRECT ROM PAGE USE
173012  012700  000400                MOV     #BIT8,R0        ; TABLE OF POSSIBLE ROM PAGES
173016  012716  000200                MOV     #200,(SP)       ; SET  POINTER TO 200 AS INITIAL PAGE BYTE COUNTER

173022  030012                  30$:    BIT     R0,(R2)         ; CHECK WHICH PAGE IS SET IN SWITCHES
173024  001003                        BNE     40$             ; BRANCH IF MATCH
173026  005721                        TST     (R1)+           ; THE TEST BIT DID NOT MATCH
173030  006300                        ASL     R0              ; CHECK THE NEXT BIT POSITION
173032  000773                        BR      30$             ; RETEST SWITCH SWTTING
                                                        ; SETUP CORRECT ROM STRUCTURE.
173034  012766  000010  000002  40$:    MOV     #10,2(SP)       ; MAKE ROM STRUCTURE = 1K.
173042  011166  000004                MOV     (R1),4(SP)      ; SET UP FOR CORRECT ROM OFFSET VALUE
173046  111137  177521                MOVB    (R1),@#PAGEHI   ; SET UP CORRECT PAGE FROM TABLE.
173052  032712  004000                BIT     #BIT11,(R2)     ; CHECK FOR DATA STRUCTURE BIT.
173056  001403                        BEQ     50$             ; BIT8= 1 THEN ROM IS A 2K STRUCTURE.
                                                                ; 2(SP) = 20 IF 2K ROM
                                                                ; 2(SP) = 10 IF 1K ROM
173060  062766  000010  000002        ADD     #10,2(SP)       ; BIT8= 0 THEN ROM IS A 1K STRUCTURE.

                                                        ; CLEAN UP FOR PROGRAM START
173066  012704  173400          50$:    MOV     #173400,R4      ; R4 = CORRECT ROM PAGE ADDRESS IN HIGH PAGE.
173072  066666  000002  000004        ADD     2(SP),4(SP)     ; MAKE 4(SP) PRESENT PAGE PLUS OFFSET.
                                                                ; THIS WILL BE USED A INDICATOR TO
                                                                ; REDO ROM ODD ADDRESSES.
                                                        ; LOOK FOR START OF INPUT FRAME.
173100  005000                  LOAD1:  CLR     R0              ; R0 = CHECKSUM REGISTER
173102  004767  000120                JSR     PC,FRAME        ; READ A BYTE FROM SELECTED ROM
173106  105303                        DECB    R3              ; R3 = DATA READ
173110  001373                        BNE     LOAD1           ; CHECK FOR +1 (START OF A BLOCK)
                                                                ; LOOP UNTIL +1 FOUND
173112  004767  000110                JSR     PC,FRAME        ; READ ANOTHER FRAME AND DISCARD.
                                                        ; GET A WORD TO BE USED AS THE BYTE COUNT.
173116  004767  000104                JSR     PC,FRAME        ; GET FIRST BYTE
173122  010305                        MOV     R3,R5           ; SAVE FIRST BYTE
173124  004767  000076                JSR     PC,FRAME        ; GET SECOND BYTE
173130  000303                        SWAB    R3              ; PUT SECOND BYTE IN HIGH POSITION
173132  050305                        BIS     R3,R5           ; NOW HAVE A WORD IN R5.
173134  010502                        MOV     R5,R2           ; R5 = WORD READ   R2 = BYTE COUNT REG.
173136  024242                        CMP     -(R2),-(R2)     ; SUBTRACT 4 TO MAKE BYTE COUNT CORRECT
                                                        ; GET A WORD TO BE USED AS THE LOAD ADDRESS.
```

```
                                ;+
                                ; ROM BOOTSTRAP
                                ;
                                ; THIS ROUTINE LOADS DATA FROM THE ROM INTO MEMORY.
                                ; THE DATA IS IN THE FORM OF FRAMES.
                                ; EACH FRAME CONTAINS 2 BYTES OF DATA.
                                ; THE FIRST FRAME CONTAINS THE LOAD ADDRESS.
                                ; THE SECOND FRAME CONTAINS THE BYTE COUNT.
                                ; THE THIRD FRAME CONTAINS THE DATA.
                                ; THE LAST FRAME CONTAINS THE CHECKSUM.
                                ; IF THE BYTE COUNT IS ZERO, THE LOAD ADDRESS IS THE JUMP ADDRESS.
                                ;-

 58 173140  004767  000062      10$:    JSR     PC,FRAME                ; GET FIRST FRAME
 59 173144  010305                      MOV     R3,R5                   ; SAVE FIRST BYTE
 60 173146  004767  000054              JSR     PC,FRAME                ; GET SECOND FRAME
 61 173152  000303                      SWAB    R3                      ; PUT SECOND BYTE IN HIGH POSITION
 62 173154  050305                      BIS     R3,R5                   ; NOW HAVE A WORD IN R5.
 63 173156  005702                      TST     R2                      ; TEST IF WORD IN R5 IS LOAD ADDRESS OR
 64 173160  001010                      BNE     20$                     ; STARTING ADDRESS. BRANCH IF LOAD ADDRESS.
 65                                                                     ; R2 IS THE BYTE COUNT REGISTER
 66                                                                     ; IF R2 = ZERO THEN THIS IS A JUMP ADDRESS.
 67
 68 173162  004767  000040              JSR     PC,FRAME                ; GET THE CHECKSUM FOR THE ADDRESS FRAME
 69                                                                     ; SAVE IT IN R3.
 70 173166  113701  177521              MOVB    @#PAGEHI,R1             ; SAVE CURRENT PAGE.
 71 173172  012737  007416  177520      MOV     #PAGE16,@#PAGELO        ; ALLOW USE OF UPPER PAGE 17 NOW.
 72 173200  000470                      BR      DOJMP                   ; R0 = CHECKSUM OF ADDRESS BLOCK
 73                                                                     ; R1 = CURRENT PAGE
 74
 75 173202  010501              20$:    MOV     R5,R1                   ; R1 = LOAD ADDRESS
 76
 77                                                             ; READ IN REMAINDER OF DATA.
 78 173204  004767  000016      LOAD2:  JSR     PC,FRAME                ; READ DATA - STORE IN R3
 79 173210  002004                      BGE     LOAD3                   ; BRANCH IF MORE DATA.
 80                                                                     ; CONDITION CODE RESULT OF
 81                                                                     ; DECREMENTING BYTE COUNT.
 82 173212  105700                      TSTB    R0                      ; CHECK IF CHECKSUM IS CORRECT.
 83 173214  001731                      BEQ     LOAD1                   ; BRANCH IF CHECKSUM CORRECT
 84
 85                                                                     ; ********************************
 86                                                                     ; CHECKSUM ERROR OCCURED.
 87                                                                     ; R0 = CHECKSUM VALUE FROM ADDING BYTES.
 88                                                                     ; R1 = ADDRS OF ENTRY INTO TABLE IF JMP FIRST INPUT FRAME
 89                                                                     ; R1 = MEMORY ADDRESS OF DATA.
 90                                                                     ; R2 = BYTE COUNT NOW = -1.
 91 173216  000000              BADLD:  HALT                            ; R3 = LAST BYTE READ. CHECKSUM READ FROM ROM.
 92                                                                     ; R4 = ADDRESS OF LAST BYTE.
 93                                                                     ; R5 = JUMP OR LOAD ADDRESS.
 94                                                                     ; PROCEEDING WILL RUN AS IF THERE WAS NO ERROR.
 95                                                                     ; ********************************
 96
 97 173220  000727                      BR      LOAD1                           ; CONTINUE  THE RUN.
 98
 99                                                             ; STORE DATA IN MEMORY.
100 173222  110321              LOAD3:  MOVB    R3,(R1)+                ; STORE DATA 8 BITS AT A TIME
101 173224  000767                      BR      LOAD2                   ; GO GET MORE DATA
```

```
 1                                                          ; GET A BYTE ROUTINE.
 2 173226  112403              FRAME:  MOVB    (R4)+,R3            ; R4 = POINTER TO DATA IN ROM.
 3 173230  005204                      INC     R4                  ; R3 = TEMPORARY STORAGE OF DATA
 4 173232  060300                      ADD     R3,R0               ; MAKE R4 POINT TO NEXT BYTE IN ROM.
 5                                                                  ; UPDATE CHECKSUM IN R0
 6 173234  042703  177400            BIC     #177400,R3          ; MASK OFF JUNK IN UPPER BYTE
 7
 8 173240  005366  000002            DEC     2(SP)               ; DECREMENT PAGE COUNT 200 OR LESS.
 9 173244  001402                    BEQ     10$                 ; BRANCH IF END OF ROM PAGE AND
10                                                                  ; CHANGE PAGE.
12                                                          ; STILL ON SAME PAGE CHECK BYTE COUNT.
13 173246  005302              5$:     DEC     R2                  ; DECREMENT BYTE COUNT BY ONE.
14 173250  000207                      RTS     PC                  ; EXIT SUBROUTINE
15                                                                  ; CONDITION CODES IS RESULT OF
16                                                                  ; DECREMENTING BYTE COUNT.
17                                                                  ; THIS IS USED BY CALLING ROUTINES.
21                                                          ; SETUP NEW ROM PAGE VALUES.
22
23 173252  105237  177521        10$:    INCB    @#PAGEHI            ; BUMP HIGH PAGE POINTER
24 173256  123766  177521  000006        CMPB    @#PAGEHI,6(SP)      ; CHECK FOR END OF ROM STRUCTURE
25 173264  001016                    BNE     30$                 ; BRANCH IF NOT AT END
27                                                          ; END OF ROM STRUCTURE.
28
29 173266  006204                    ASR     R4                  ; TEST R4 ADDRESS IF EVEN OR ODD.
30 173270  103405                    BCS     20$                 ; IF ADDRESS IS ODD END OF ROM STRUCTURE.
31 173272  146637  000004  177521    BICB    4(SP),@#PAGEHI      ; ADDRESS EVEN RESET ORIGINAL PAGE
32 173300  000261                    SEC                         ; SET CARRY TO INDICATE ODD ADDRESS
33                                                                  ; STILL TO DO.
34 173302  000410                    BR      35$                 ; AND DO ODD BYTES.
35
36                                                          ; R4 IS ODD DO NEXT ROM STRUCTURE
37 173304  113766  177521  000006  20$:    MOVB    @#PAGEHI,6(SP)      ; R4 ADDRESS WAS ODD END OF ROM STRUCTURE
38 173312  066666  000004  000006        ADD     4(SP),6(SP)         ; MAKE ROM STRUCTURE PRESENT VALUE PLUS OFFSET.
39 173320  000401                    BR      35$                 ; THE ADD WILL CLEAR CARRY FOR
40                                                                  ; CORRECT ENTRY INTO 35$
42                                                          ; COMMON EXIT
43 173322  006204                30$:    ASR     R4                  ; TEST PRESENT ROM ADDRESS IF ODD OR EVEN.
44                                                                  ; TEST WILL BE DONE ON CARRY BIT LATER.
45
47                                                                  ; TEST CARRY TO DETERMIND IF EVEN OR ODD.
48 173324  012704  173400        35$:    MOV     #173400,R4          ; MAKE R4 RESTART ADDRESS.
49 173330  103001                    BCC     40$                 ; IF CARRY SET MAKE IT ODD ADDRESS IN R4.
50 173332  005204                    INC     R4
52                                                          ; SET PAGE COUNT
54 173334  012766  000176  000002  40$:    MOV     #176,2(SP)          ; UPDATE ROM PAGE COUNTER.
55 173342  132737  000001  177521        BITB    #BIT0,@#PAGEHI      ; TEST FOR EVEN OR ODD PAGE
56 173350  001003                    BNE     45$                 ; BRANCH IF ODD
57 173352  062766  000002  000002        ADD     #2,2(SP)            ; PAGE IS EVEN MAKE BYTE COUNTER 200
```

```
     58 173360  000732                  45$:    BR      5$                      ; RETURN
     59                                                                          ; IF PAGE IS EVEN PAGE COUNT = 200
     60                                                                          ; IF PAGE IS ODD PAGE COUNT = 172
     61                                                                          ; ODD PAGE LAST TWO LOCATION ARE SKIPED
     62                                                                          ; THERE USE IS RESERVED FOR DIAGNOSTIC.
     63
     64                                                              ; CHECK IF A JUMP OR HALT AT END
     65 173362                          DOJMP:
     66 173362  105700                          TSTB    R0                      ; TEST CHECKSUM  0 = CORRECT CHECKSUM
     67 173364  001403                          BEQ     10$                     ; BRANCH IF CHECKSUM IS OK.
     68
     69                                                      ; ********************************************************
     70 173366  000000                          HALT                            ; CHECKSUM ERROR ON ADDRESS BLOCK
     71 173370  000167  177404                  JMP     ROBOOT                  ; PROCEEDING FROM HALT WILL RESTART ROM BOOTSTRAP
     72                                                                          ; R0 = RESULTING CHECKSUM SHOULD BE ZERO
     73                                                                          ; R1 = CURRENT PAGE OF EXECUTION
     74                                                                          ; R2 = BYTE COUNT SHOULD BE -1.(177 777)
     75                                                                          ; R3 = LAST BYTE READ FROM ROM
     76                                                                          ; R4 = ADDRESS OF THE LAST BYTE READ.
     77                                                                          ; R5 = ADDRESS TO JUMP TO.
     78                                                                          ; R6 = STACK POINTER
     79                                                              ; ************************************************
     80
     81 173374  032705  000001          10$:    BIT     #BIT0,R5                ; CHECK LOW ORDER BIT.
     82 173400  001403                          BEQ     DOJMP1                  ; BRANCH IF LOAD ADDRESS IS EVEN
     83
     84
     85
     86                                                      ; LOAD ADDRESS ODD ERROR HALT.
     87
     88                                                                          ; ******************************
     89 173402  000000                          HALT                            ; LOAD ADDRESS ODD
     90 173404  000167  177370                  JMP     ROBOOT                  ; CONTINUE WILL RESTART ROM BOOT
     91                                                                          ; R0 = CHECKSUM RESULT = 000 000
     92                                                                          ; R1 = LAST PAGE USED TO EXECUTE FROM.
     93                                                                          ; R2 = BYTE COUNT NOW = -1.(177 777)
     94                                                                          ; R3 = LAST BYTE READ.
     95                                                                          ; R4 = ADDRESS OF DATA BYTE.
     96                                                                          ; R5 = JUMP TO ADDRESS.
     97                                                                          ; ******************************
     98
     99 173410                      DOJMP1:
    100 173410  112737  000037  177524          MOVB    #BLANK,@#LIGHTS                 ; CLEAR THE LIGHTS
    101 173416  110137  177520                  MOVB    R1,@#PAGELO                     ; CHANGE PAGE LOW VALUE
    102 173422  000115                          JMP     (R5)                            ; JUMP TO STARTING ADDRESS
```

```
                                                                        ; SET ROM ROUTINE
                
173424                          SETROM:
                
173424  012702  177524                  MOV     #LIGHTS,R2      ; ADDRESS OF LIGHTS AND SWITCH IN R2
173430  012712  000005                  MOV     #TWELVE,(R2)    ; TURN LIGHTS ON = 12.
                
                                                ; FIND UPPER MEMORY LIMIT.
173434  012706  000004                  MOV     #4,SP           ; MAKE STACK POINTER = 4
173440  012726  173004                  MOV     #FOUND,(SP)+    ; RETURN ADDRESS WHEN NON-EXISTANT MEMORY FOUND
173444  012716  000340                  MOV     #340,(SP)       ; CLEAR THE PSW OF INTERRUPT ENABLE
                
173450  011626                  10$:    MOV     (SP),(SP)+      ; TEST FOR NON-EXISTANT MEMORY BY READ EACH LOCATION
173452  000776                          BR      10$             ; AND WRITING IT BACK,THEN INCREMENT THE POINTER
                
173454  000020                  TABL11: PAGE20                  ; PAGE FOR ADDITIONAL DIAGNOSTIC 2K ROM
173456  000040                          PAGE40                  ; PAGE FOR EPROM 2K
173460  000200                          PAG200                  ; PAGE FOR 16K SYSTEM ROM
```

```
                                .SBTTL  CHANGE PAGE ROUTINE FOR PAGE SEVENTEEN.
                                ;* CHANGE PAGE LOCATIONS FOR THIS PAGE.
                                ;* R4 WILL CONTAIN THE NEW PAGE SETTING.
                                ;* R5 WILL CONTAIN THE NEW PC.
                                ;* AFTER THE NEW PAGE IS LOADED INTO THE PAGE CONTROL REGISTER, THE
                                ;* MOV  R5,PC WILL BE EXECUTED FROM THE NEW PAGE.
                                ;* LOCATION 173 776 = RESERVED FOR FUTURE DIAGNOSTIC USE.
                                ;* LOCATION 173 777 = CHECKSUM FOR THE TWO PAGES.
                                ;*

                                        ;FILL REMAINING ROM LOCATIONS WITH HALTS.

 173462  000000                 .WORD   000000
 173464  000000                 .WORD   000000
 173466  000000                 .WORD   000000
 173470  000000                 .WORD   000000
 173472  000000                 .WORD   000000
 173474  000000                 .WORD   000000
 173476  000000                 .WORD   000000
 173500  000000                 .WORD   000000
 173502  000000                 .WORD   000000
 173504  000000                 .WORD   000000
 173506  000000                 .WORD   000000
 173510  000000                 .WORD   000000
 173512  000000                 .WORD   000000
 173514  000000                 .WORD   000000
 173516  000000                 .WORD   000000
 173520  000000                 .WORD   000000
 173522  000000                 .WORD   000000
 173524  000000                 .WORD   000000
 173526  000000                 .WORD   000000
 173530  000000                 .WORD   000000
 173532  000000                 .WORD   000000
 173534  000000                 .WORD   000000
 173536  000000                 .WORD   000000
 173540  000000                 .WORD   000000
 173542  000000                 .WORD   000000
 173544  000000                 .WORD   000000
 173546  000000                 .WORD   000000
 173550  000000                 .WORD   000000
 173552  000000                 .WORD   000000
 173554  000000                 .WORD   000000
 173556  000000                 .WORD   000000
 173560  000000                 .WORD   000000
 173562  000000                 .WORD   000000
 173564  000000                 .WORD   000000
 173566  000000                 .WORD   000000
 173570  000000                 .WORD   000000
 173572  000000                 .WORD   000000
 173574  000000                 .WORD   000000
 173576  000000                 .WORD   000000
```

CHANGE PAGE ROUTINE FOR PAGE SEVENTEEN.

```
     59 173600  000000                .WORD   000000
     60 173602  000000                .WORD   000000
     61 173604  000000                .WORD   000000
     62 173606  000000                .WORD   000000
     63 173610  000000                .WORD   000000
     64 173612  000000                .WORD   000000
     65 173614  000000                .WORD   000000
     66 173616  000000                .WORD   000000
     67 173620  000000                .WORD   000000
     68 173622  000000                .WORD   000000
     69 173624  000000                .WORD   000000
     70 173626  000000                .WORD   000000
     71 173630  000000                .WORD   000000
     72 173632  000000                .WORD   000000
     73 173634  000000                .WORD   000000
     74 173636  000000                .WORD   000000
     75 173640  000000                .WORD   000000
     76 173642  000000                .WORD   000000
     77 173644  000000                .WORD   000000
     78 173646  000000                .WORD   000000
     79 173650  000000                .WORD   000000
     80 173652  000000                .WORD   000000
     81 173654  000000                .WORD   000000
     82 173656  000000                .WORD   000000
     83 173660  000000                .WORD   000000
     84 173662  000000                .WORD   000000
     85 173664  000000                .WORD   000000
     86 173666  000000                .WORD   000000
     87 173670  000000                .WORD   000000
     88 173672  000000                .WORD   000000
     89 173674  000000                .WORD   000000
     90 173676  000000                .WORD   000000
     91 173700  000000                .WORD   000000
     92 173702  000000                .WORD   000000
     93 173704  000000                .WORD   000000
     94 173706  000000                .WORD   000000
     95 173710  000000                .WORD   000000
     96 173712  000000                .WORD   000000
     97 173714  000000                .WORD   000000
     98 173716  000000                .WORD   000000
     99 173720  000000                .WORD   000000
    100 173722  000000                .WORD   000000
    101 173724  000000                .WORD   000000
    102 173726  000000                .WORD   000000
    103 173730  000000                .WORD   000000
    104 173732  000000                .WORD   000000
    105 173734  000000                .WORD   000000
    106 173736  000000                .WORD   000000
    107 173740  000000                .WORD   000000
    108 173742  000000                .WORD   000000
```

CHANGE PAGE ROUTINE FOR PAGE SEVENTEEN.

```
    109 173744  000000                .WORD   000000
    110 173746  000000                .WORD   000000
    111 173750  000000                .WORD   000000
    112 173752  000000                .WORD   000000
    113 173754  000000                .WORD   000000
    114 173756  000000                .WORD   000000
    115 173760  000000                .WORD   000000
    116 173762  000000                .WORD   000000
    117 173764  000000                .WORD   000000
    118         173766                .=173766
    120 173766  010437  177520        CPAG17: MOV     R4,@#PAGELO             ; LOAD THE NEW PAGE WORD
    121 173772  010507                        MOV     R5,PC                   ; JAM THE NEW PC
    122
    123 173774  000000                        .WORD 000000                    ; RESERVED FOR FUTURE DIAGNOSTIC USE
    124
    125 173776  000000                        .WORD 000000                    ; PAGE CHECKSUM VALUE SUPPLIED
    126                                                                       ; BY ROM BLASTING PROGRAM.
    127
    128         000001'               .END
```